User Guide Version 6.2

Volume 3: POST-PROCESSING

“Resolution of partial differential equations is more about art than science“.

Apocryphal quotation from Numerical Recipes in Fortran

“2 + 2 = 4 except for large values of 2“

Anonymous

“42“

Douglas Adams

Edited by:

MICRESS group

Contents

1 How to use this manual 5

2 Introduction 6

3 Post-Processing 8

3.1 Introduction to MICRESS® Post- Processing 8

3.1.1. Output file types 8

3.1.1 “mcr” files 9

3.1.2 ASCII/ txt files 11

3.1.3 vtk files 13

3.2 Overview of post-processing tools for MICRESS® results 14

4 Running DP_MICRESS 15

4.1 Quick start bar options 16

4.1.1 Time settings 17

4.1.2 Coordinate system choice 17

4.1.3 Mouse pointer tools 18

4.2 The task bar options 19

4.2.1 File 19

4.2.2 Window 20

4.2.3 Content 24

4.2.4 Repition and mirroring 24

4.2.5 Annotations 29

4.2.6 Animation 31

4.2.7 Virtual EDX 31

4.2.8 Mouse pointer 33

4.2.9 Grouping 33

4.2.10 Scripting 35

4.2.11 Options 47

4.3 Hotkey chains 48

5 ParaView 55

5.1 Installing and running ParaView 55

5.2 Using ParaView 56

5.2.1 Display surface or volume 56

5.2.2 Rotation/ Translation 56

5.2.3 Colour by different results and attributes 56

5.2.4 Run an animation 57

5.2.5 Applying filters/setting thresholds/making sections 57

6 gnuplot 59

6.1 Installing gnuplot 59
6.2 Plotting data and fitting curves 60
6.3 Comparing different data 62

7 Data conversion: *.mcr to *.vtk 63

8 Post Processing: Exercises 64

8.1 Introduction to post-processing exercises with DP_MICRESS 64
8.2 Read and handle a single result file for a single time step 64
8.3 Read and handle a single result file for multiple time steps 67
8.4 Read multiple result files 68

9 Useful Information and FAQs 72

9.1 Why can I not open a certain 3D output file by DP_MICRESS? 72
9.2 How to calculate the*.intf-output 72
9.3 How to combine phase boundaries and concentration in DP_MICRESS 73
9.4 How to save an animation as a direct output in DP_MICRESS? 73

10 References 74

11 Literature 75

1 How to use this manual

The "Introduction" shortly presents the capabilities of the software as well as its wide range of applications. It also introduces the recent developments and improvements of MICRESS®.

The chapter "Post-Processing" offers an overview of the MICRESS® result files and presents the freeware that can be used for processing, analyzing and visualisation of MICRESS® output.

The capabilities of the programs DP_MICRESS, ParaView and gnuplot are discussed in the subsequent chapters. Information is given about how to install these software tools and how to use their graphical interface. The chapters also provide some examples of possible evaluation and visualisation that may be helpful when post-processing MICRESS® results.

Some "Post-processing exercises" may be helpful to get better acquainted with the evaluation of MICRESS® results. You may find further useful hints for generating specific output and troubleshooting in "Useful Information and FAQ".

Any suggestions or comments for improvements on the MICRESS® manual and documentation are highly appreciated and welcome. Please send an e-mail to documentation@micress.de.

2 Introduction

The software MICRESS® (MICRostructure Evolution Simulation Software) is developed for time- and space-resolved numerical simulations of solidification, grain growth, recrystallisation or solid state transformations in metallic alloys. MICRESS® covers phase evolution, solutal and thermal diffusion and transformation strain in the solid state. It enables the calculation of microstructure formation in time and space by solving the free boundary problem of moving phase boundaries.

Microstructure evolution is governed essentially by thermodynamic driving forces, diffusion and curvature. In case of multicomponent alloys, the required thermodynamic data can either be provided to MICRESS® in the form of locally linearised phase diagrams, or by direct coupling to thermodynamic data sets via a special TQ interface, developed in collaboration with Thermo Calc™ AB, Stockholm.

MICRESS® is based on the multiphase-field method which defines a phase-field parameter for each phase involved. The phase-field parameter describes the fraction of each phase as a continuous function of space and time. Each single grain is mapped to a distinct phase-field parameter and is treated as an individual phase. A set of coupled partial differential equations is formed which describes the evolution of the phase-field parameter, together with concentration, temperature, stress and flow fields. The total set of equations is solved explicitly by the finite difference method on a cubic grid.

2D and 3D simulations are possible. The size of the simulation domain, the number of grains, phases and components is restricted mainly by the available memory size and CPU speed.

Telling us suggestions for improvements on the manual or commenting on the manual is possible by writing an e-mail to documentation@micress.de.

MICRESS® handles:

- 1-, 2- and 3-dimensional calculation domains
- arbitrary number of components, phases and grains
- solid-solid and solid-liquid interactions
- anisotropy of grain boundaries, mobility and energy

MICRESS® supports:

- coupling to thermodynamic databases (via the TQ-interface of Thermo Calc™)

In the present MICRESS® Vol.3 "post processing" you will find:

- an overview of the different MICRESS® result files
- an overview of the available software for post-processing of MICRESS® result files
- a short overview of the functions of DP_MICRESS, ParaView and gnuplot
- examples of possible data post-processing
- some useful information, troubleshooting and frequently asked questions

A description of the phase-field phenomenology and theoretical background can be found in MICRESS Vol. 0: Phenomenological Background. MICRESS Vol. I: MICRESS Installation provides information about the installation of the software and explains how to use the software with the help of an example. MICRESS Part III: Post-processing explains the possibilities the user has while treating output results. MICRESS Part IV: Examples contains a collection of examples for simulations performed on various topics.

3 Post-Processing

3.1 Introduction to MICRESS® Post- Processing

Before starting post-processing, make sure that the files to be analysed are the most recent ones you created. It is often useful to check whether MICRESS® has really started creating the output immediately after starting the simulation. For this purpose it is often wise to compare the computer time and the time of file creation/update in the results directory.

Once you have verified the right output files, you should decide about the best of post-processing the individual files according to their contents.

The different output file types are shortly highlighted in the following.

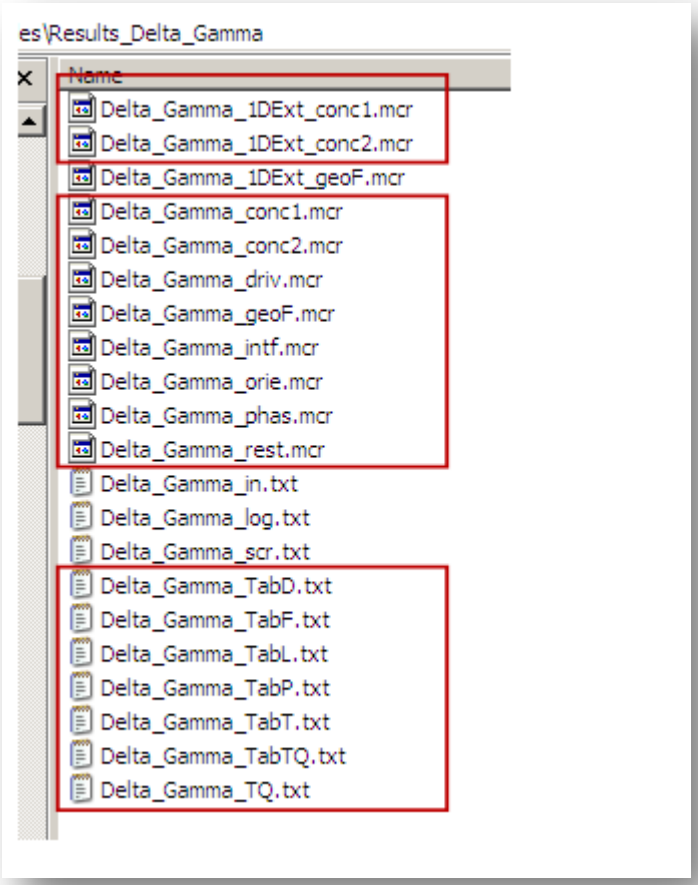


Figure 3.1 An example of MICRESS® output files

3.1.1. Output file types

MICRESS® generates the following categories of result files:

- files containing field values, the ".mcr" files
- ASCII ".txt" files containing statistical information
- optional: "vtk" files (especially for 3 D results)

3.1.1 "mcr" files

Three different classes of MICRESS result files (".mcr") can be distinguished:

- special .mcr files
- standard .mcr files
- 1d_extension .mcr files

Special mcr files:

output	Description
*geoF.mcr	a geometry file being necessary for any post-processing. It stores information concerning the geometry of the simulation domain e.g. grid size, cell dimensions, etc. This is file is used by any post-processing program for the visualisation of results.
*rest.mcr	a restart file that allows to restart/continue a simulation.

Results for the individual field variables (2D, 3D) for all output time steps in standard ".mcr" files:

output	Description
*.concN.mcr	gives information about the concentration of the alloying element N
*.cPhaN.mcr	similar to *.concN, except for the interface regions, where the concentration in a given phase is written out and not the mixture concentration
*.dGsp.mcr	used in stress-coupled simulations; it shows the stress contribution to the driving force
*.diff.mcr	shows the data provided by Thermo-Calc™ via its TQ interface for the diffusion coefficients
*.driv.mcr	gives the driving force for the growth of a grain with a higher number into a lower one
*.frac.mcr	yields the fraction of solid
*.fracN.mcr	shows the fraction of the phase N
*.geoF.mcr	stores information concerning the geometry (i.e.: grid size, cell dimension)
*.hStr .mcr	provides information about the hydrostatic stress, when the stress coupling is activated
*.intf .mcr	provides data fro tracking the interface
*.korn.mcr	a graphical output for the grain field, containing the numbers assigned to grains in a consecutive manner during their definition in the initial grain structure or their nucleation during runtime
*.krum.mcr	gives information about the interface curvature

*.millx.mcr	also *.milly and *.millz lists the respective Miller indices
*.mueS.mcr	gives information about the local interface mobility
*.orie.mcr	stores the 2D-grain orientation
*.phas.mcr	provides information on the phases in the system
*.rest.mcr	contains data necessary to restart a simulation; this binary file is for internal use by MICRESS® and cannot be post-processed
*.rex .mcr	lists the stored energy
*.sxxCV.mcr	also *.sxyCV, *.sxzCV, *.syyCV, *.syzCV and *.szzCV components of the stress tensor, xxCV, (yyCV), zzCV are the diagonal elements in the 2D (3D) case, sxyCV, sxzCV, syzCV are off-diagonal terms (torsion), respectively. The output is available when stress coupling is activated.
*.temp.mcr	For the temperature in Kelvin, when temperature coupling is activated. It can be also used for plotting temperature-distance profiles.
*.uxCV.mcr	also *.uyCV and *.uzCV. They record the normal displacements when the stress coupling is activated.
*.vel.mcr	gives out the interface velocity
*.vxCV.mcr	also *.vyCV, *.vzCV, *.syyCV, *.syzCV and *.szzCV: components of the stress tensor; xxCV, (yyCV), zzCV are the diagonal elements in the 2D (3D) case, sxyCV, sxzCV, syzCV are off-diagonal terms (torsion), respectively. This is true if the stress coupling is activated.
*.vM.mcr	gives out the von Mises stress when stress coupling is activated.

Results for individual field variables calculated in extended 1D fields (1DExt):

Output	Description
*1DExt_conc1.mcr	shows the concentration of alloy element 1 as specified in the MICRESS driving file, when a 1D external field has been used in the simulation
*1DExt_conc2.mcr	shows the concentration of alloy element 2 as specified in the MICRESS driving file, when a 1D external field has been used in the simulation
*1DTemp_Temp.mcr	gives the temperature when a 1D temperature field is used
*1DTemp_enth.mcr	shows the enthalpy when a 1D temperature field is used

3.1.2 ASCII/ txt files

Two different classes of MICRESS ASCII result files (".txt") can be distinguished:

- special .txt files
- standard .txt files

The meaning and content of these files is described in the following tables:

Special ".txt" output files:

output	Description
*in. txt	the input file corresponds to the driving file as understood by MICRESS®.
*log.txt	gives information about the input data, initial relinearisation output at initialisation or before the first nucleation, CPU time and intermediate output at defined time steps

".txt" files containing dedicated data:

output	**Description**
*.TabC.txt	for the average and extrema of concentration in each phase, when the concentration coupling is activated
*.TabD.txt	gives the diffusion coefficients at the corresponding "tab_log" times, when the concentration coupling is activated
*.dTLat.txt	used for intermediate data relative to the release of latent heat
*.TabF.txt	for the average fraction of each phase in the whole domain, when the concentration coupling is activated
*.TabGD.txt	contains information about the grain status, i.e. whether a grain increases in dimensions
*.TabK.txt	contains the number of grains. This file is updated by default each time a grain is set or disappears.
*.TabL.txt	an ASCII-monitoring output generated at user-defined intervals to check the simulation progress; it gives information on the temperature gradient and some additional information depending on the type of coupling activated.
*.TabLin.txt	it gives the temperature at defined time intervals for the temperature at the bottom and at the connecting points when linear flow trend used

*.TabN.txt	stores data relevant for making "Von Neumann – Mullins plots"
*.TabP.txt	sums up the simulation time, the CPU time, the diffusion, the seed, the enthalpy, the stress and the temperature times
*.TabR.txt	provides information on the recrystallized fraction
*.TabO.txt	gives orientation information
*.TabT.txt	reports information concerning the setting of the (automatic) time-step, when automatic time-stepping has been selected
*TabTQ.txt	shows the temperature change with progressing simulation time and the corresponding CPU time
*TQ.txt	a Windows-specific data file which stores the output of Thermo-Calc´s interface making it more convenient to exploit the origin of TQ errors. It is always created under Windows when Thermo-Calc™ -coupling is activated

3.1.3 vtk files

Especially for 3D simulations, e.g. Fig. 3.2, there is an option to save simulated data in a special format called "vtk". This format allows 3D visualisation of the results in freeware software tools like e.g. paraview (www.paraview.org; see chapter 5).

This output option can be switched on in the driving file for the simulation. As a consequence, "vtk" files are written out instead of "mcr" files. The major difference between these two output formats is, that a "mcr" file contains the results for one field variable for all time steps, while each "vtk" file comprises all field variables for only one time step.

A conversion of "mcr" files to "vtk" files is possible using the DP_MICRESS tool (see chapter 7).

Figure 3.2 3D Simulation of a an array of Mg dendrites with the results being saved and displayed in the vtk format

3.2 Overview of post-processing tools for MICRESS® results

Currently available freeware frequently used for post-processing different MICRESS® result files comprises:

- *DP_MICRESS*. This is a 2D (partially 3D) evaluation tool, the graphical user interface to Display MICRESS. This tool is provided along with your MICRESS® installation CD.
- *ParaView.* This is a freely available 3D post-processor from ParaView (www.paraview.org)
- *gnuplot.* This command-line driven plotting utility can be used for post-processing of text and ASCII files. It can be downloaded from www.gnuplot.info. The ASCII and txt result files can be also post-processed with any type of ASCII post-processing, e.g. MS Editor, Excel, etc.

software	2D	3D	text	ASCII
DP_MICRESS	☐	☐		
gnuplot			☐	☐
ParaView		☐		

Figure 3.3 Categorisation of the currently available freeware for post-processing of MICRESS® results

These tools and their basic operation will be shortly introduced in the following chapters.

4 Running DP_MICRESS

Display MICRESS (short name: DP_MICRESS) is a post-processing tool with a wide span of possibilities, ranging from basic display and image output, to advanced features such as zooming, multiple display, build-in symmetry options, animations and virtual EDX analysis. It can handle several files simultaneously. DP_MICRESS is preferentially used for reading and post-processing 2D and3D MICRESS® output files. Nearly all of the functionality of the graphical user interface is scriptable for batch processing.

It reads result files in binary (single and double precision, native or not) and ASCII-format, both compressed and not-compressed. Moreover, it performs conversions between these formats. This post-processing tool can be freely downloaded from the download area at www.micress.de.

Display MICRESS can also be called from a shell, providing the name of the result file(s) to be displayed. It can be called by entering e.g. DP_MICRESS Delta_Gamma_intf.mcr in the command line where the name DP_MICRESS serves here as a place holder for the name of the current binary for the software, e.g. DP_MICRESS_7.1.2_x64. The software also takes a DP_MICRESS script as a command line argument (see chapter 4.2.10 for scripting).

Windows user can simply “drag and drop” result files to the DP_MICRESS icon or double click on the result files to open them. DP_MICRESS always initially shows the result for the first time step. Use the buttons “next” and “last” to view the results for the next time steps.

The graphical user interface is explained in detail in the following chapters, starting up with a brief visual summary explaining the start window (See Fig. 4.1).

The quick start bar is described, followed by a more detailed perspective explaining the entire spectrum of the available functions.

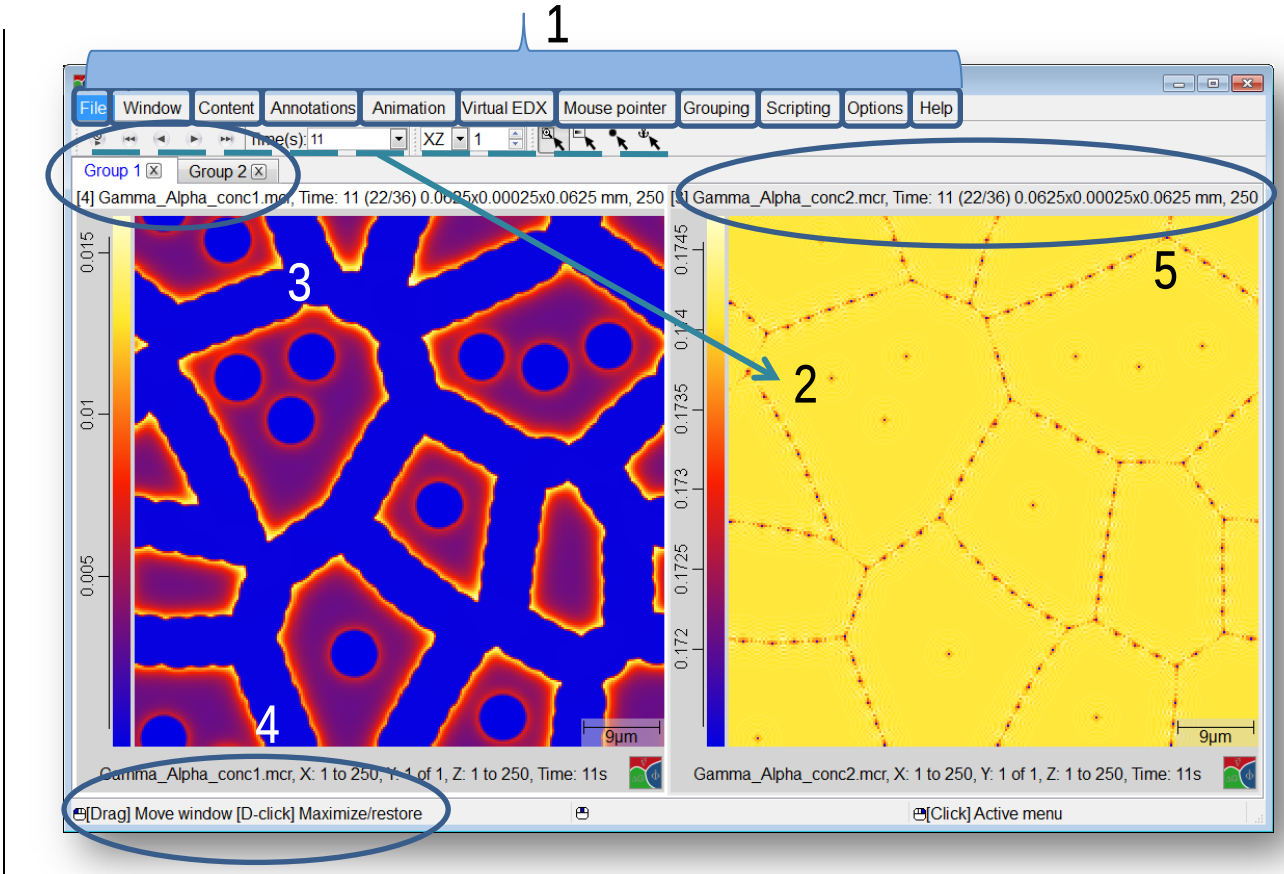


Figure 4.1 The DP_MICRESS start window

1	The taskbar enables all the functions to modificate the active result files
2	The quick start bar contains some central, frequently used applications of the taskbar
3	Several groups can be created and though selected as being active. So, different result files are observable.
4	The proceeded hotkeys are shown.
5	Presenting a data file with it's characteristics and current classifications above and below the simulation window provides a clear overview on the simulating status quo.

4.1 Quick start bar options

All taskbar options are applied to the active windows of the actual group. By right-clickling on the window bar a group member can be either activated or deactivated. Beyond, there are the options controlling the activating of other groups or the deleting the currently chosen dataset. When moving the curser over several fields, information about the possible actions and applications, in relation to the field on which one is moving, are monitored in a special information bar. This information bar can be seen at the window's botton.

4.1.1 Time settings

There are several tools to take a look at the specifically desired animation´s time steps. By using the field on the left, the whole simulation can be watched fluently from the first step to the last step of the animation (1). Moreover, it is possible to move forward and backwards in time (3) and to switch to the start or end of the simulation (2). The program can move to the first or last time step (2). In case of putting the some files of the same calculations into execution, the last time step is replaced by the latest common time step. For instance, R1: 1-5 and R2:1-10 output on a first click the 5th, and on a second click a 10th step. And consequently there is also a step to the following, or previously time step (3) (see Figure 4.3). Likewise, it is also possible to choose a certain time [s] manually (see Figure 4.2)

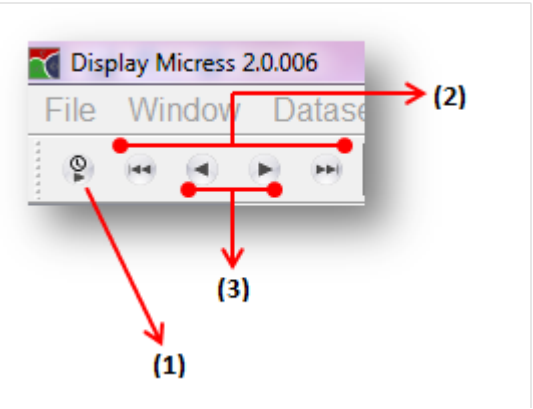


Figure 4.2 The animation's different time step options

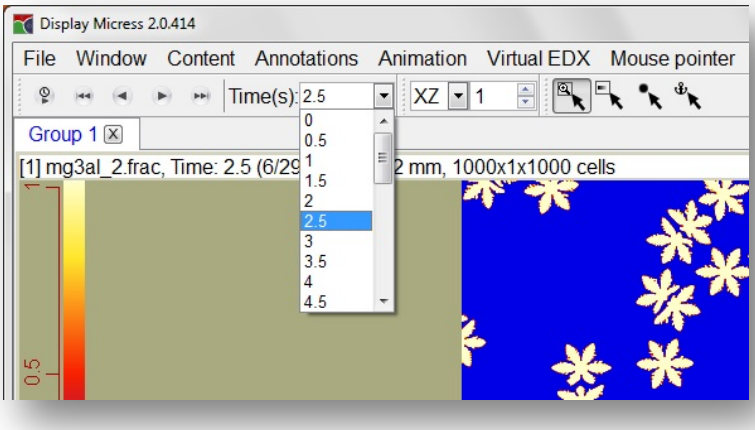
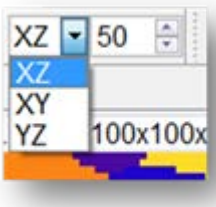


Figure 4.3 Manual time setting

4.1.2 Coordinate system choice



3D data sets can only be displayed in 2D axis parallel slides whereas XZ, XY, YZ define the visible dimensions and the reference unit the position in the 3rd dimension.The 2D can be done by selecting the coordinate system together with its reference unit. So the reference value can be a number from 1 to 75 moving virtually through the datasets volume. As Figure 4.4 reveals, the mid Y location of the XZ plane takes the value 50 in a 100x100x100 dataset- A more detailed description of this principle can be found in the introductory passage of Section 4.4 'Content'.

Figure 4.4 Coordinate system

4.1.3 Mouse pointer tools

There are a few functions highlighted for quickly using these in a handy manner (see Figure 4.4). Apart from highlighting these tools in the task bar, the same tools can be found in the rubric 'Mouse pointer'.

The zooming tool allows a closer observation of certain areas by just moving the cursor in any direction and hence opening an extract of a certain area. The prerequisite is here to zoom within the open simulation. It is possible to move the visible area on the same zoom level by holding the 'ctrl'-key while dragging. The zoomed in region can be cancelled out by clicking once. If having moved as well, a simple clicking switches back to the originally zoomed in area again. By clicking a second time the zooming is reversed, too.

The rectangle drawer enables a change in scaling and also makes use of these values for the color scale's minimum and maximum values. A simple clickling reactivates the automatic settings of the color scale.

The points and lines tool reveals the properties of one single point in a label. When adding another point, the two points are automatically connected by a line. This line generates a direct comparison by seeing the distance and the angle between the two points. By clicking twice on a certain point, it will be removed. The transparency of the labels depends on the cursor's position: just when clicking on a point the label is readable. So when creating a new point the other points become transparent. Lines and points can be moved by dragging them. When removing a line, the two related lines at the two ends are connected to each other automatically. A double-clicking on a line removes the point-line combination. All created points are deleted by a second click on the points and lines tool. The selected points and lines can be used as input for the virtual EDX functions, for more detail see chapter 4.7. In spite of zooming, the selected points keep constantly their position.

The anchor gives a reference value to another point. One click places the anchor and to define the anchor's position it can be moved to a desired point. Beyond, by clicking on the arrowhead the anchor's target position can still be changed. A double-clicking on the arrowhead activates a popping-up window which comprises several designing opportunities and moreover the possibility of deleting the arrow. Even after zooming, the selected anchors keep constantly their positions.

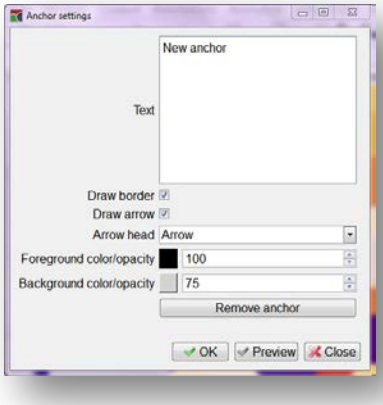


Figure 4.5 The mouse pointer tools

4.2 The task bar options

4.2.1 File

A MICRESS® results file (see Chapter 3.1.1) can be opened with DP MICRESS, which can be both a new data file by 'Open solution' or a beforehand used one by 'Load recent'. Several files can be bundled into a group so that commands are done for all the windows belonging to the group. This tool facilitates working with DP MICRESS®.

The simulation results can be saved in several data formats for other usages, p.e. by the 'Export group as image' application (See Fig 4.6). More detailed information on the 'Export window contents as dataset' application you will find in Chapter 7 'Data conversion: *.mcr to *.vtk'.

In Figure 4.6 the settings for an image export are presented. There are several image formats available: The tif-format is especially recommendable when dealing with uncompressed files, but consumes also a lot of storage space because of the high quality standard. Also, the selection can be restricted to the actual needs. This means, the numer of used timesteps or the 3rd dimension. Moreover, the picture quality can be changed in general by modifying the resolution.

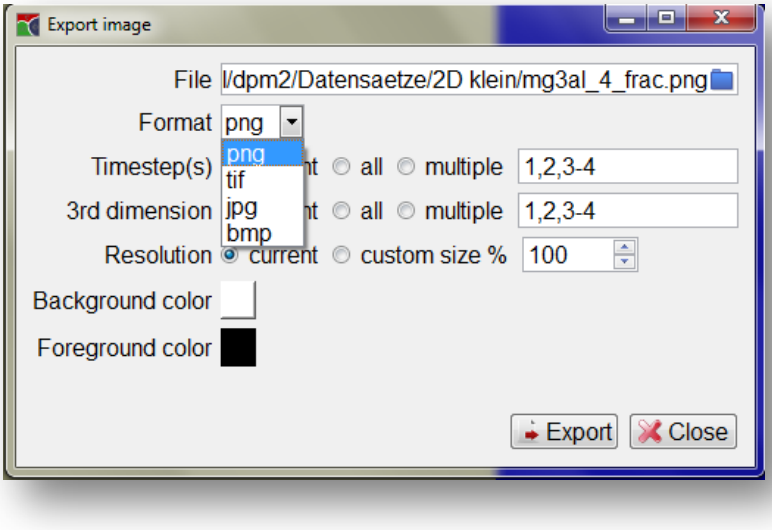


Figure 4.6 The formats and options for saving the result files as images

The format of which is made use for a MICRESS data file, is the MICRESS® standard format. 'VTK' is a format used by Kitware's visualisation toolkit (VTK) and is very common in scientific image processing and visualisation. There are two versions available: On the one hand binary is a compact, non-human-readable data format. On the other hand, exporting results as ASCII data can be used for small data sets to check the results manually. An uncompressed ASCII data files consumes much more disk space than raw binary data.

When exporting `window contents as dataset' there are several options available:

- `Region' specifies the selected area's scope. Choosing the entire revealed area or limiting the export is correspondingly possible. The limitation can for example derive from the zooming function or from turning a 3d file into a 2d file by selecting only two of the XYZ coordinates.
- `Geometry' specifies the file's size, with the unit `cells' in grid cells. The data file has default settings, which can thus be converted into custom settings. Only when using the geometry variation, the tool `geometry conversion' is applied additionally.

When exporting `group as image' only the image resolution, and not the content itself, is changed when selecting anything else than `current'. To improve the image resolution is useful especially when dealing with big data sets. To export the currently visible window, the file must be renamed to assure not overwriting the original file. Modifying more than the `current' settings in relation to both timestep and 3d leads to some obligatory variations: In case of multiple time steps, a star (*) in the output file name will be taken as a place holder for the time step. Respectively a question mark (?) for multiple 3D planes should be added. Usually these variations are set automatically. When changing one's mind after putting this variation into execution and hence going back to `current', the star or question mark have to be removed manually when considered disturbing. `Resolution' varies the presenting of the group's size. `Background color' and `Foreground color ' change the background an text color of the exported image.

4.2.2 Window

This section deals with the arrangement of the windows, which are active datasets opened as one group by DP_MICRESS. There can be created several groups which is described in section 'Grouping' 4.2.9. The selection can be removed by pressing 'Close all windows`. Right-clicking on the window bar opens an context menu which provides additional action, e.g. closing a dat set window. Double-clicking maximizes, resp. minimizes, the data set window.

4.2.2.1 Tile arrangement

This tool refers to the field in which an active dataset can be revealed. So DP_MICRESS makes use of a grid layout. This grid's size can be varied by the Tile arrangement tool. Either `width' or `height' are varied, the other values are adjusted automatically according to the fitting window size. Applying this is done by Tile windows (See the next passage).

Tile arrangement settings become especially useful when needing to arrange a group of several datasets because a group's windows can be manually positioned. For the application by default, a grid layout is used but with the Tile arrangement, the grid dimensions can be set individually.

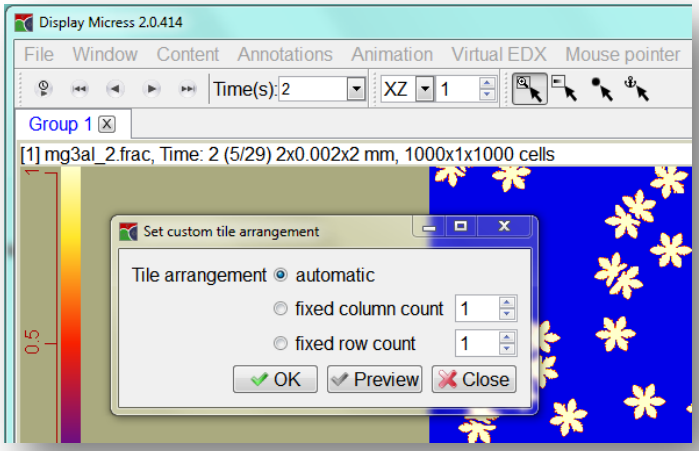
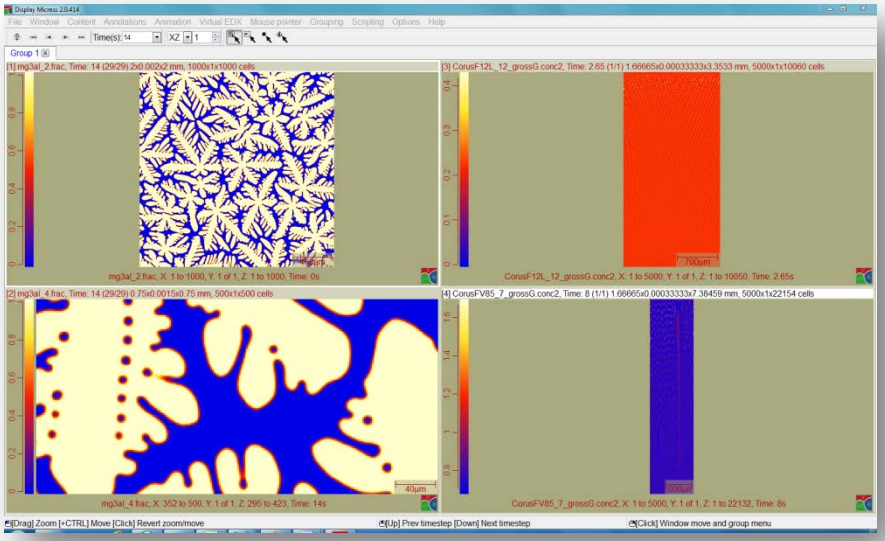

Display Micress 2.0.414
File Window Content Annotations Animation Virtual EDX Mouse pointer
Time(s): 2
XZ 1
Group 1
[1] mg3al_2.frac, Time: 2 (5/29) 2x0.002x2 mm, 1000x1x1000 cells
Set custom tile arrangement
Tile arrangement automatic
fixed column count 1
fixed row count 1
OK Preview Close

Figure 4.7 Tile arrangement

4.2.2.2 Tile windows



The data sets are organized in a way that the available space is fully utilized and consequently the background is avoided. So in order to set DP_MICRESS in terms of individual needs, it is possible to arrange the windows by the **Tile windows** tool, applying the selected grid layout. The window size is calculated depending on the main window's size (See Figure 4.8).

Figure 4.8 Tile windows

4.2.2.3 Cascade windows

Cascade windows also refers on one active group, like all tools exept of “Fit main windows size”. The size of the active group is reduced and the active datasets are at the same time ordered in a way that the file’s titles can be seen, but only one dataset can be seen completely as top window. Clicking on the file’s properties switches to another top window. (See Figure 4.9)

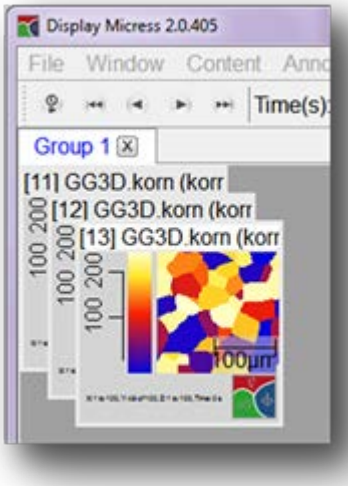


Figure 4.9 Cascade Windows

4.2.2.4 Window sizes

There are three possibilities for manipulating the window size, also for all the windows in one group. ‘Set window sizes’ enables a manual input in height and width in the unit pixel. This function is not applied for the whole window, but for every dataset separately. So a dataset in one active group needs to be chosen first, and then can be varied (See Figure 4.9).

Figure 4.10 Set window sizes

‘Fit window sizes’ automatically takes the geometric form of a dataset into account by adjusting the appropriate format. The superior aim is minimizing the space to take a look at the whole active data set (See Figure 4.10).

‘Fit main window size’ modifies the size of the MICRESS main window in a way to make the currently active group as clearly visible as possible, hence minimizing the background space, as possible (See Figure 4.11).

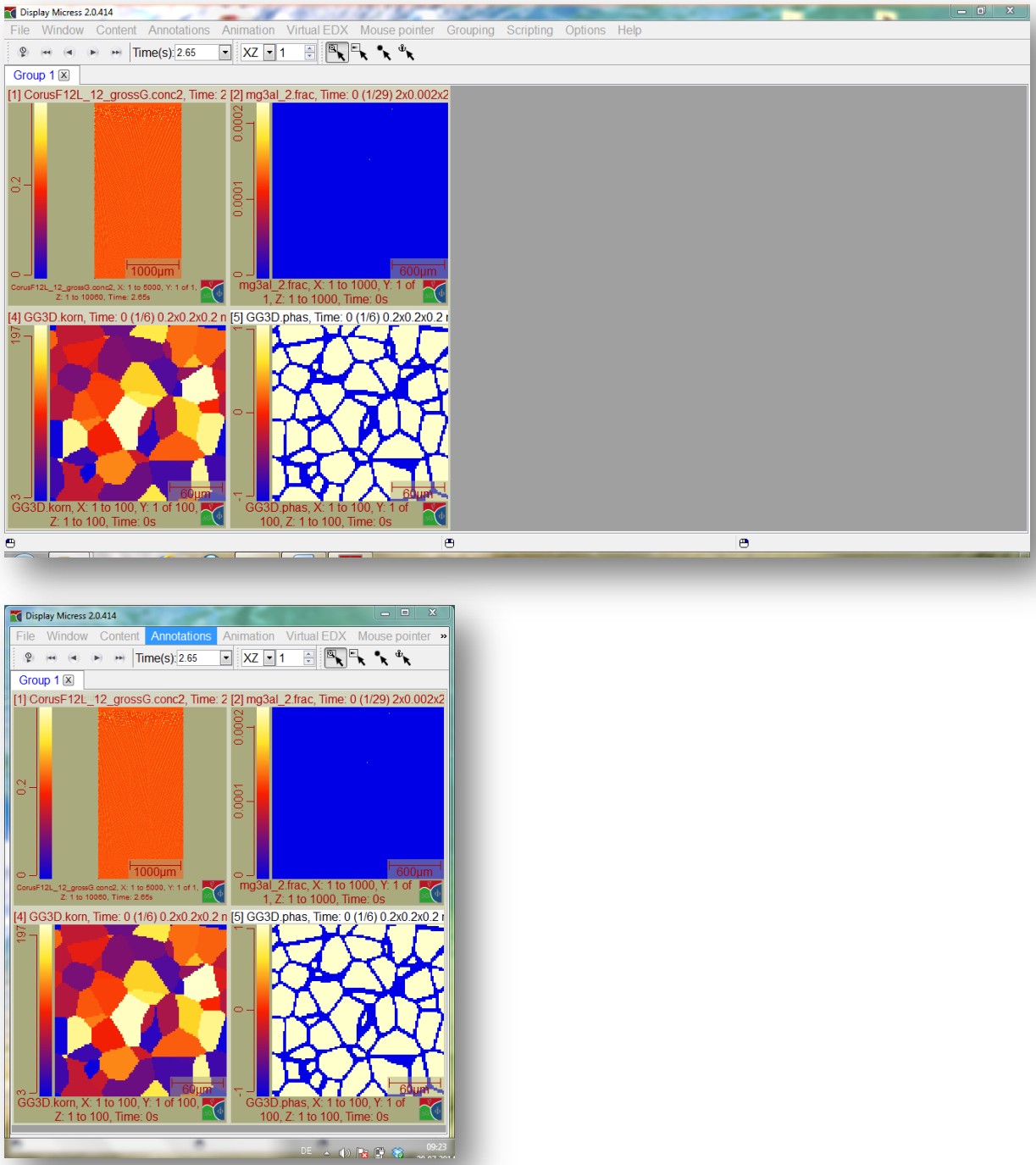

Display Micress 2.0.414
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting Options Help
Time(s): 2.65
XZ 1
Group 1
[1] CorusF12L_12_grossG.conc2, Time: 2
[2] mg3al_2.frac, Time: 0 (1/29) 2x0.002x2
1000µm
600µm
[4] GG3D.korn, Time: 0 (1/6) 0.2x0.2x0.2 n
[5] GG3D.phas, Time: 0 (1/6) 0.2x0.2x0.2
60µm
GG3D.korn, X: 1 to 100, Y: 1 of 100, Z: 1 to 100, Time: 0s
GG3D.phas, X: 1 to 100, Y: 1 of 100, Z: 1 to 100, Time: 0s
mg3al_2.frac, X: 1 to 1000, Y: 1 of 1, Z: 1 to 1000, Time: 0s

Figure 4.11 Fit window sizes (above) and fit main window (below)

4.2.2.5 Duplicate

This tool creates an exact copy of the dataset's current time step and in general of the current settings, in the same group.

4.2.3 Content

These tools change the perspective from which the simulation is observed. Moreover they enable a manipulation of several parameters, depending on the user's needs. Based on all applications is the idea of watching a certain unit, for example a cell: Moving between e.g. 1 and 75 depending on data set where every number is a layer of a cross-section. This works by moving virtually through the cell's volume. When entering 1 the front is revealed, the higher the numbers become, the distance to the front gets bigger. Finally the number 75 stands for the backside of the cell. A coordination system can be chosen here. `Cutting plane up´ and `Cutting plane down' enables a scrolling through the several layers.

4.2.4 Repition and mirroring

This tool refers on one chosen active dataset and duplicates the dataset in a specific manner. So firstly there has to be decided whether this tool is applied horizontally or vertically. For both, the same operations can be executed. Then, the dataset can be repeated by stringing the same files together, twice or three times. This is the Repetition function. The Mirroring tool works alike: The same dataset is mirrored in a series twice or three times. This form of expression can be revised by pressing the `disabled' button. All the options are visualized in Figure 4.12..

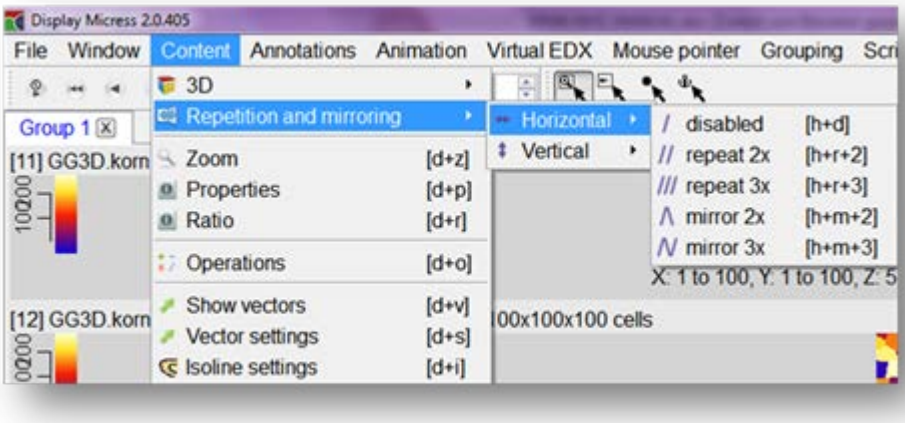


Figure 4.12 Repetition and Mirroring

4.2.4.1 Zoom

Principally the Zoom tool serves the same purpose as the zooming by opening a rectangle. It just differs from the rectangle zooming in the manually setting. Figure 4.13 shows the possible adjustments. The unit of reference are cells. When zooming the rest of the dataset is normally cut. Here DP_MICRESS makes use of a special function: If there is still space for the cut area after zooming, this area is kept on showing until the 'hide outside datapoints' field is activated.

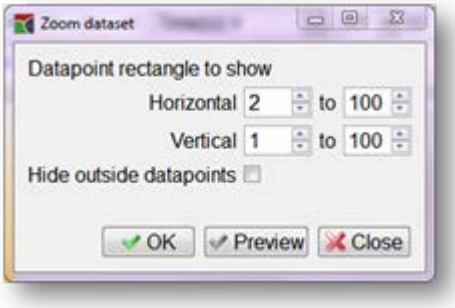


Figure 4.13 Zoom

There is a special case regarding the Virtual EDX: If a Virtual EDX line is marked, it is not restricted to the zoom, but can also be outside the zoom. This occurs when selecting `Zoom in'. Analogously,

the Default Line Scan becomes invalid when ‘Repetition’ or ‘Mirroring’ are selected.

4.2.4.2 Properties

By clicking on the Properties rubric, statistical characteristics are pointed out, such as:

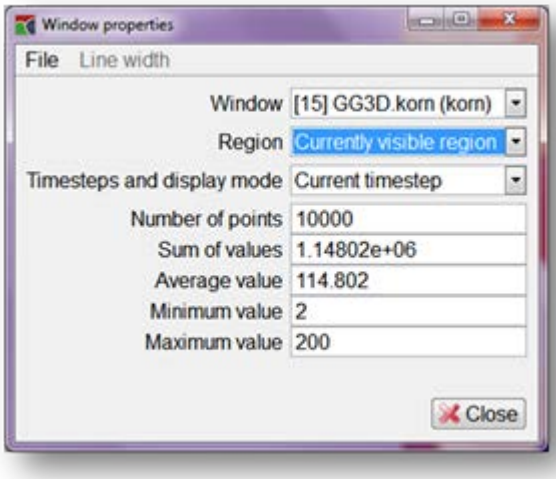


Figure 4.14 Properties

Window: Here the dataset about which statistical information are given is defined.

Region: Chosing the visible area of interest. So if the whole dataset shell be watched, or only a limited version of the dataset. In case of zooming there is no difference in the image section because the currently visible region will tally with the zooming region.

Timesteps and display mode: whether the actual timesteps or all timesteps can be selected. Consequently, a window is popping up revealing a visualization as graph or table. This graph can be varified by setting several characterictics in a desired format. For example the lines can be set differently in their color or thickness. Exporting the results as table is only possible when visualizing the graph and not as table beforehand.

4.2.4.3 Ratio

This application gives the ratio of width to height. So it is possible to set this ratio in a different way by entering the data according to an adequate presentation. This function is especially useful for one-dimensional datasets or cell sizes, which are not programmed in a 1/1 ratio.

Figure 4.15 Ratio

4.2.4.4 Operations

As a prerequisite for the practical ‘Operations’- tool, there has to be selected or also created a window on which the operations are applied. Here, also more than one dataset can be selected by pressing the ‘Plus’- button. In this way the number of utilized datasets will be increased. The dataset, which should be manipulated, needs then to be specified by selecting the utilized datasets explicitly. The maximum number of possible datasets is 22.

Basically, this function comprises a mathematical variation of a dataset. To each dataset belongs a parameter appearing as a letter. This parameter serves for a comparison between several datasets and is used entering mathematical commands. For example it can be squared. Clicking on 'Help' and then on 'Command reference' leads to an online list of possible calculations, which can be realized using DP MICRESS. It is important to know that these applications do not only consist of simple mathematical modules but also of mathematical functions like sine function or like connections from the field of logics.

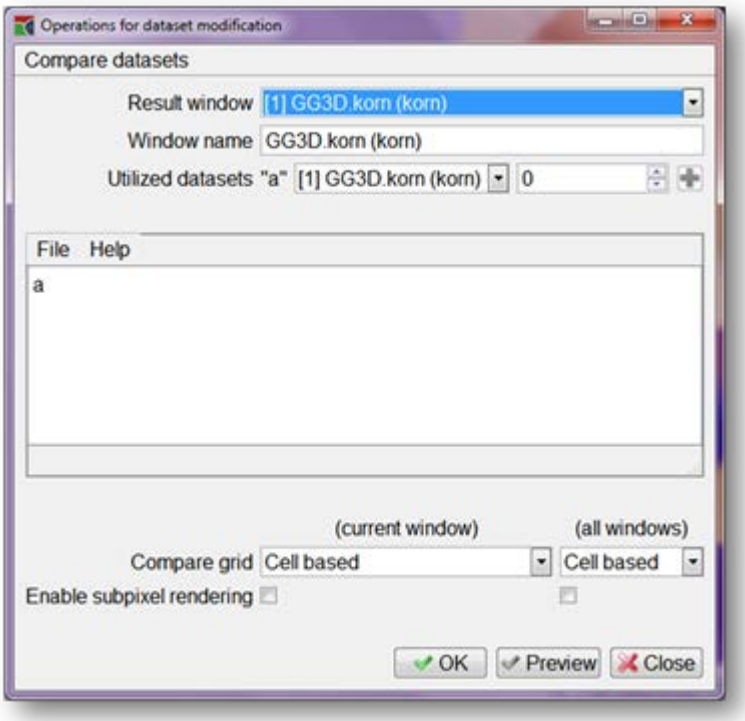


Figure 4.16 Operations

The letters x,y,z, and t are not part of a possible parameter selection since they are already unambiguously defined variable names for the cell's coordinates [millimeter] and time [s].

On the field 'File' the created operations can be exported by selecting 'Save' or imported into another data file by selecting 'Load'.

It is possible to watch the same data file from different time step perspectives: So the past in respect of the current time step, are negative values, the future is revealed hence by positive values.

Different datasets can be compared using several parameters either on a 'Cell base' or a 'Scale base'. This comparison works on the level of choosing all the group's windows and just some of them, depending on individual demands. The standard operation for calculating a comparison between two datasets is a simple subtraction, such as a-b.

'Sub pixel rendering' is a tool which refers on the showed pixels: When it is activated, all dataset's pixels with their underlying pixels are shown which enables sharper borders and clearer shapes. When the tool is deactivated, only the pixel in the left upper corner is shown including all underlying ones.

4.2.4.5 Show vectors

This tool is especially appied when working with vector files. So the arrows stand for velocities in x or z direction. Negative values go to the left, hence positive values to the right. The z-coordinate influences the way an arrow is pointing up or down. Moreover, an arrow's length indicates its strength. The longer the arrow, the stronger is the current.

The line mode deals with flow patterns by simulating a single particle's movements.

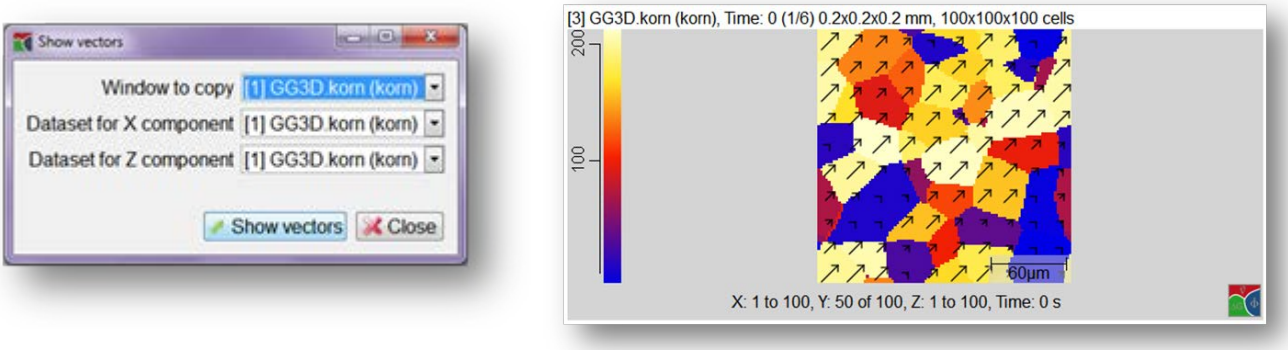


Figure 4.9 Show vectors

4.2.4.6 Vector settings

When setting the ´Cutoff for absolute minimum length' tool there is chosen a minimum length which is presented. Every vector being shorter than this length is filtered out.

When setting the `Cutoff for absolute maximum length' tool there is chosen consequently a maximum length. Every vector being longer than this length is presented as the maximum length.

Grid size expresses the distance between two arrow centers in the unit image pixels.

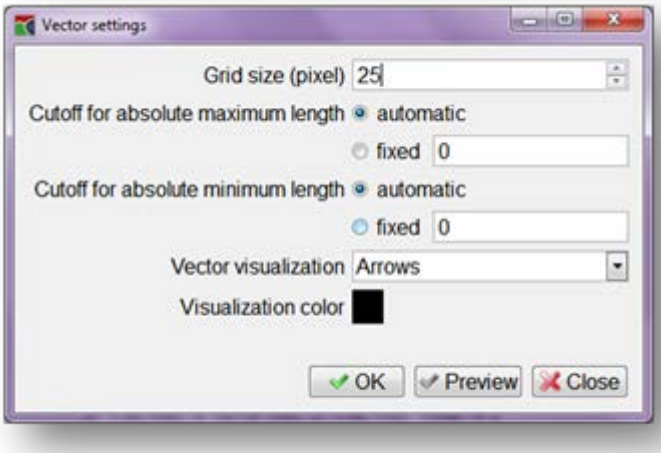


Figure 4.10 Vector settings

4.2.4.7 Isoline settings

Isolines refer on areas, in which the same values prevail. So, in DP_MICRESS they reveal for example consistent concentration values or 0.5 phase fraction line as a representative of the diffuse interface.

It is possible to draw multiple isolines in one data set window. The respective iso value for each line can be chosen manually or by calculating an äquidistant value for the currently displayed area of the data set by '**normalize**'.

The '**Window/dataset relation**' allows to define in which window an isoline of the used dataset should be drawn. A variation in this relation enables for example a 0.5 phase fraction isoline in a concentration filed. In Figure 4.8, a 0.5 percent isoline will be drawn in the first window 'Delta_Gamm.conc2' using the dataset of the second concentration because this is the content of the current window.

'**Line width**' and '**Background** transparency' define common properties for all line. The coloring of each line can be set by the squared buttons beside the iso value. The left one defines the inner color of the line, the right the color of a thin border of the line.

'**Preserve timesteps**' enables a multiline plot over time, i.e. while stepping through the time steps, already drawn lines will be preserved.

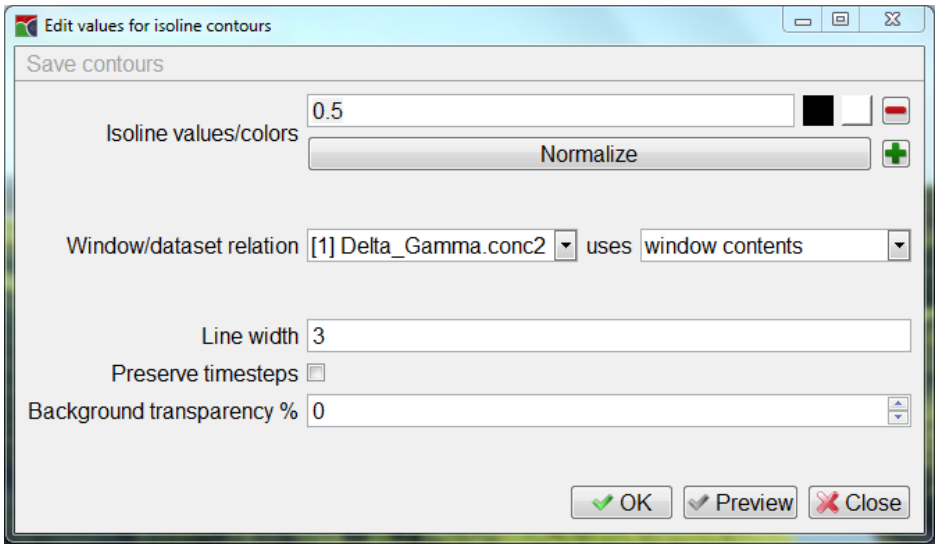


Figure 4.8 Isoline settings

4.2.5 Annotations

4.2.5.1 Color Scale

The color scale can be setted in several ways: It can be hidden, the color presenting can be changed and by clicking on 'Position' it can be moved to a fixed position on the top, to the left/right, and on the bottom. 'Independent' stands for the moving of the color scale to a separately variable dataset.

'Flip scale' defines, whether the coordination system's scale is located on the right or on the left of the color scale itself.

The distance between color scale and window boundary can be varied by using the tool 'Set gap'. There, a window pops up, asking to insert a number in percent, which stands for the portion of actually used space in reference on the window's boundary. So if this percentage is increased, the gap between color scale and window boundary is enhanced likewise. As a consequence, the color scale's width decreases.

By clicking on 'Set range', the color scale's spectrum with minimum and maximum value can be chosen. Referring on the current timestep, the current region or all timesteps can be selected. By clicking on 'Set size' the color scale can be widened or narrowed .

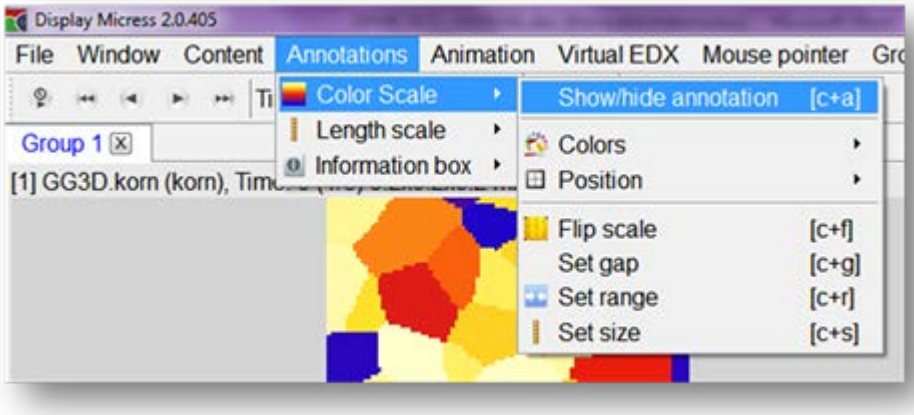


Figure 4.20 Color Scale options

4.2.5.2 Length Scale

The Length Scale rubric contains settings concerning the unit showing the datasets (See Figure 4.21).

By using the tool 'Show/hide annotation' can be chosen wether the length scale, monitored on the right hand side lower corner, in the simulation is revealed or not (See Figure 4.22). This annotation can be horizontally or vertically aligned by using the clicking on `Horizontal' or `Vertical'.

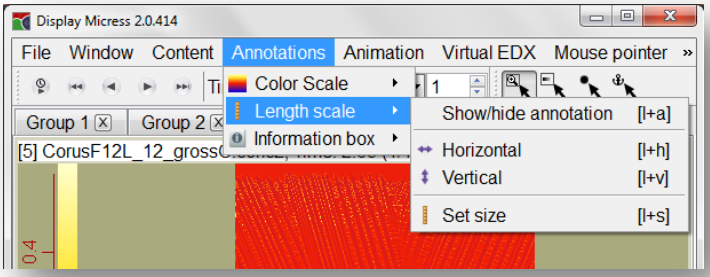


Figure 4.21 Length scale settings

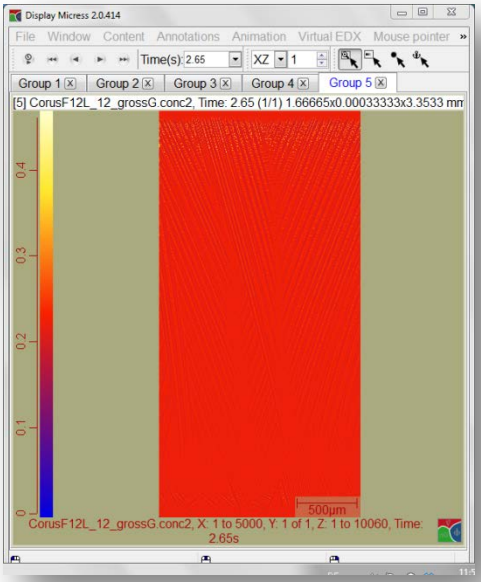


Figure 4.22 The length scale is active

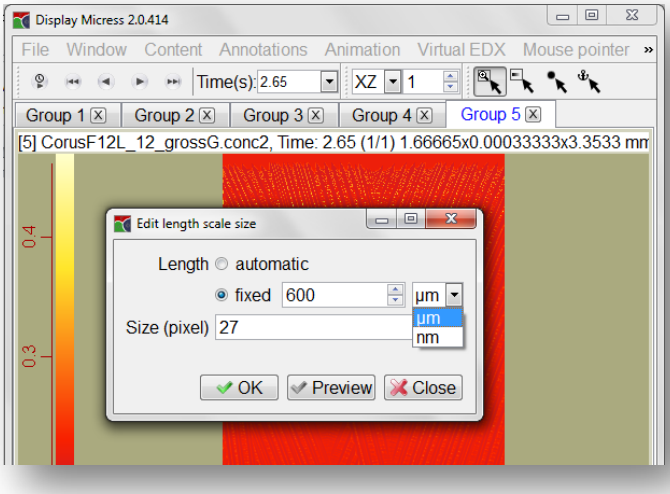


Figure 4.23 Edit length scale size

The `Edit length scale size' option enables, besides the automatically setted, selecting a different scale within the data file. The observable file segment itself stays the same, only the length scale's extension is varied. Additionally, the length scale's unit can be chosen between micro- and nanometers. Furthermore, in the `Side' field the space, the length scale comprises can be reduced or enlarged

(See Figure 4.23).

4.2.5.3 Information Box

The information box generates or hides the data file's properties and its status. There are several options, such as the location for these specific information. So the position of the information box can be defined by simply clicking on the field where the bar is marked in bold (Right, Top, Left or Bottom).

`Set content' The information line which will be displayed in the information box can be edited. Several dynamic keywords are available, e.g. *$w* for the current window name.

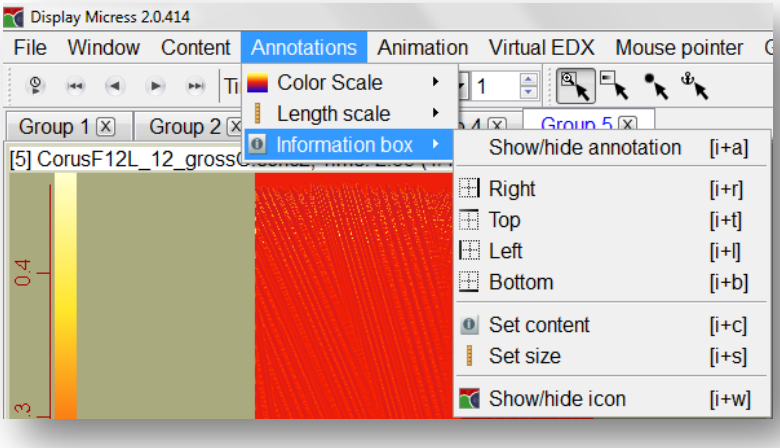


Figure 4.24 The information box content

`Set size' can analogously to the length scale options reduce or enlarge the field, in which the information are provided. The MICRESS logo can be hidden or shown as required.

4.2.6 Animation

The animation tool has got the same application as the task bar option 4.1 `Time settings' and is just a second way of dealing with different time steps.

4.2.7 Virtual EDX

The energy of X-ray lines are indicators for classifying elements, therefore also an alloy's composition. These lines can be simulated by DP_MICRESS.

- Pick default line: lines can be added by a simple clicking.
- Pick points/lines with mouse: This tool has got the same application as the task bar option points and lines tool in the rubric 4.1.3 Mouse pointer tools.
- Pick points and lines with coordinates: an exact indicating of the coordinates where points and lines shall be created is possible

The lines and points can be removed alike the taskbar options.

`Field cell': This tool comprises the same application as the MICRESS cell pointer, where 3d coordinates are transformed into 1d coordinates.

`Show line data' makes it possible to take a look at the pathway of the values on the line, going from the starting point to the line's end point. In the popping up default window, the line width can be modified manually which is then valid for all selected graphs. This window is applicated in all edx functions and can be enlarged or decreased by sliding it. Conveniently, there can be zoomed in the graphics, and zooming out is done consequently by clicking. One row represents a line, and when having selected a row this is verified by a tick. Removing a specific line from the graph presentation can be simply managed by unchecking the corresponding row, hence this is reversed by clicking on the row a second time. By double clicking on the file's designation manipulating the graph's color is possible. In the field `Time' a specific time step can be selected. In contrast to the tool of unchecking a row, a removing of a graph by clicking on the `x' is definite. Reactivating the graph requires a second file loading.

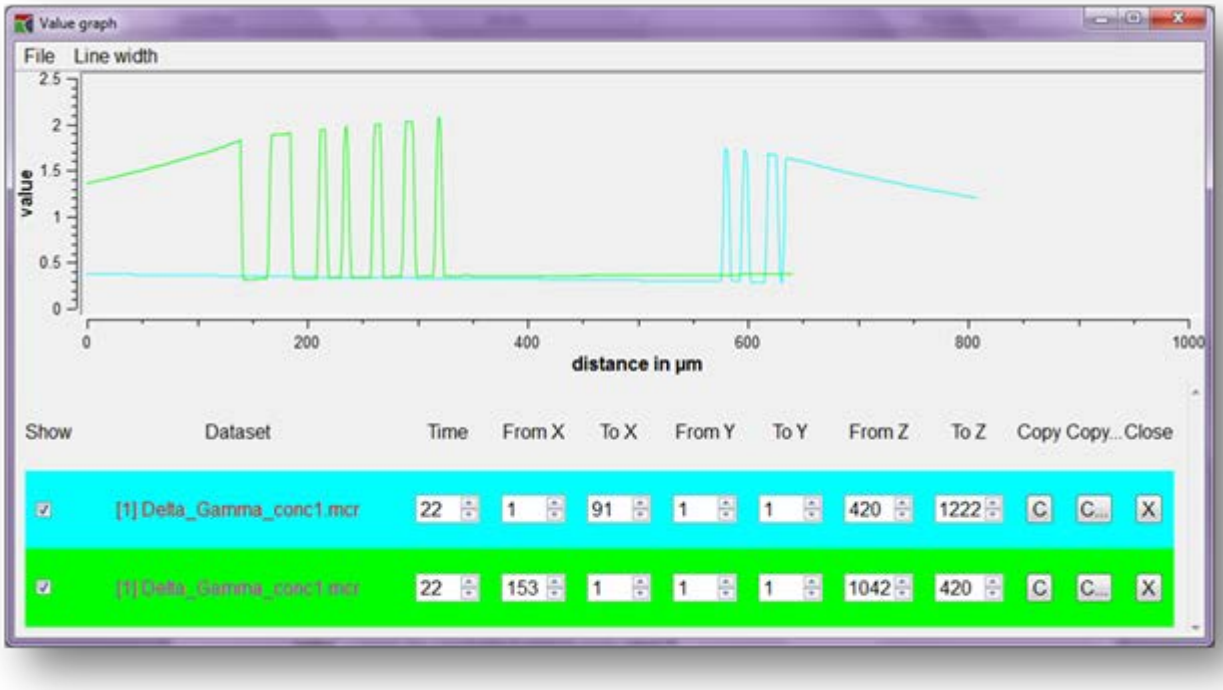
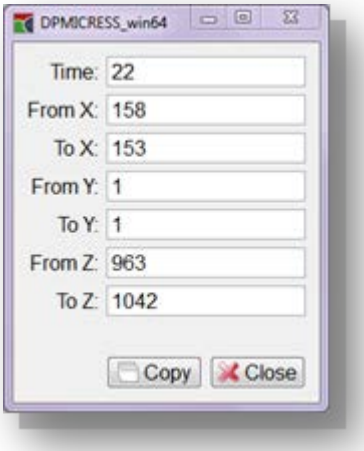

Value graph
File Line width
value
distance in µm
Show Dataset Time From X To X From Y To Y From Z To Z Copy Copy... Close
[1] Delta_Gamma_conc1.mcr 22 1 91 1 1 420 1222 C C... X
[1] Delta_Gamma_conc1.mcr 22 153 1 1 1 1042 420 C C... X

Figure 4.25 The line data value graph

The ‚Copy' tool creates an exact copy of the present data file. The ‚Copy…' tool works alike, except of the possibility to vary the copy according to a user defining beforehand. For a visualization of the possible settings, see Figure 4.26.



The basic idea of copying comprises creating a copy for each single value.

(If there are not used more than two windows,) a data file can be pulled from one onto another window by moving it's row. Despite the fact that being empty after pulling away the whole window's data, and hence being temporarily empty, the window stays and be used again. So there can be created several line data windows for the same data file, running simultaneously and can be assembled or separated again as needed.

.

Figure 4.26 The 'Copy…' settings

The`File' field enables exporting the data as table. If there are several graphs, there is created a sequence of columns. So the x and y values of each graph are placed behind each other (See Figure 4.27). `Don't export value' contains the opportunity to exclude some values.

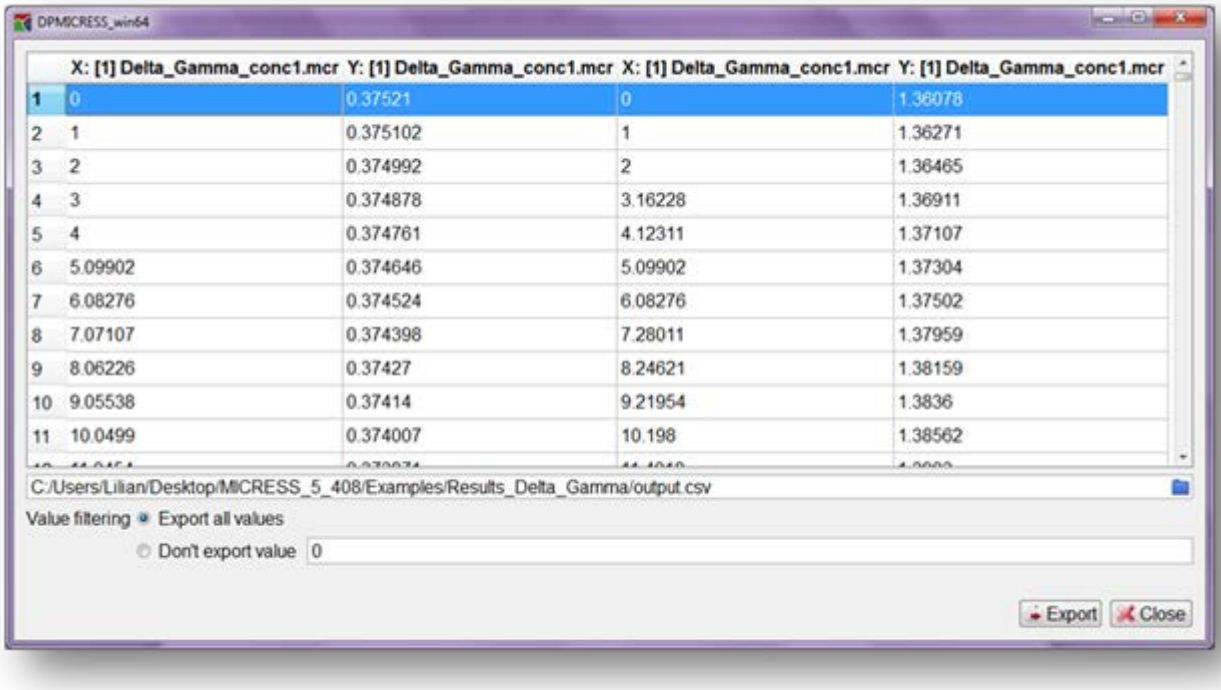


Figure 4.12 Exporting the data file of Figure 4.24's graph as a table

`Show Line data over time' shows the line data in reference to all time steps, where one line symbolizes one time step.

`Show Point data over time' works just as `Show Line data over time' only dealing with points instead of lines. Each point belongs to one line. So on the graph there is not seen the lines' course but the points' course over the time. This lack of spatial variance is the reason why it is only possible to define one setting for XYZ.

`Show value over time' is entering and therefore searching manually a defined value on a line. The program responds this request by letting the graph reveal where exactly this value is achieved. The search refers on the distance to the first point of a line.

`Show non zero values' monitors every value above or beyond zero. This tool is convenient for data sets which consist especially of zero values, to point out the deviating values.

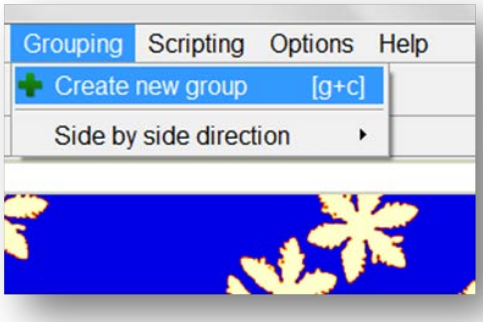
4.2.8 Mouse pointer

This tool has got the same application as the so-called task bar option, the rubric 4.2.1.3 Mouse pointer tools.

4.2.9 Grouping

'Grouping' enables moving the tabs in order to hide a group or to present the groups next to each other. The boundary between tabs can be moved to vary an extract. Creating a group, which contains several datasets, entails a central benefit: Using the DP_MICRESS tools affects *all the files* of one group. So an effective comparison between datasets of the same status is realized. Concerning moving and arranging of a group, the handling is improved by the 'Drag & Drop' tool.

A practical application is the double-clicking on the dataset's labeling: The selected dataset is enlarged and its window opens full screen.



Starting up with creating a group, they can be arranged in many ways (See Figure 4.28).

The boundary between tabs can be moved to vary an extract. `Side by side direction' changes the way of presenting a group in way of either being horizontally or vertically arranged.

Figure 4.28 Options for grouping

Furthermore, as shown in Figure 4.29, there are several options available when dealing with groups: By right clicking on the information bar of a specified dataset (marked in green) a dataset can be activated so that it is marked for further editing. Analougously it can be deactivated for excluding it from further editing. 'Close dataset' removes the dataset from the DP_MICRESS window, and virtually deletes the files for the present preparation. Another option is the 'Activate/ Deactivate other datasets' field, where the other datasets within one group can be controled.

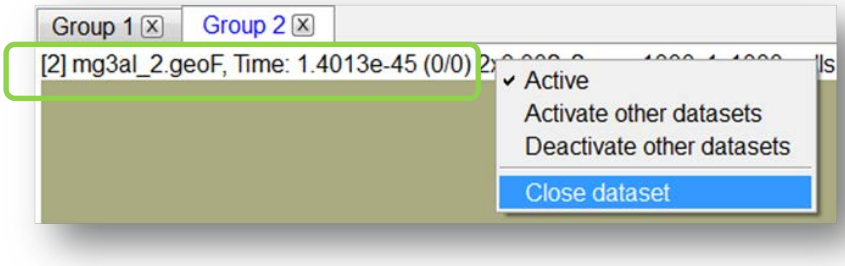


Figure 4.29 Activating, deactivating and deleting datasets within a group

The marked group's designation, on which hot keys are then applied, is colored in blue and can thus be identified. If one window is marked, the entire group is automatically marked as being active. Nevertheless, a single dataset, which is part of a group, can still be activated or deactivated independently. If a dataset is deactivated, its label switches from black into red (See Fig. 4.30)

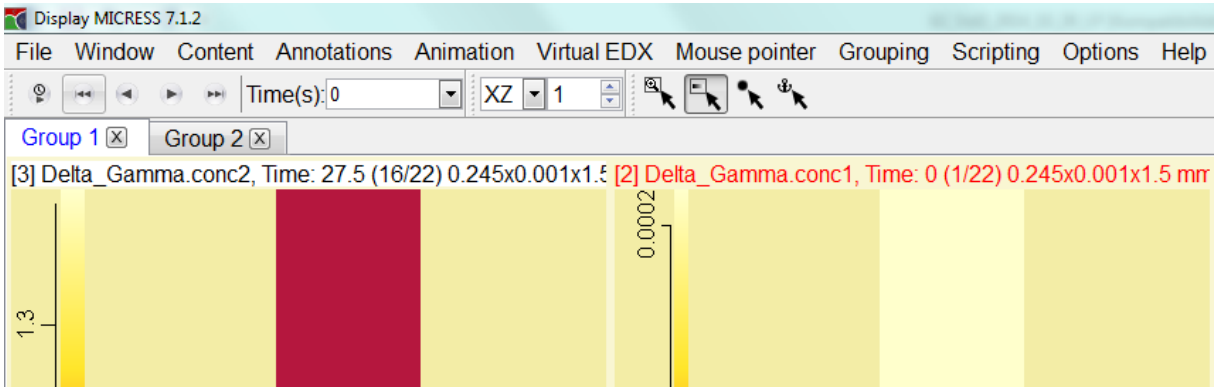


Figure 30 Active for editing is dataset [3] (as its label is being colored in black), part of Group 1 (as ists label is colored in blue)

In Figure 4.31 a vertical arranged presentation of two groups is presented. Taking the red arrow into account, here you can see that the border between the two groups is moveable.

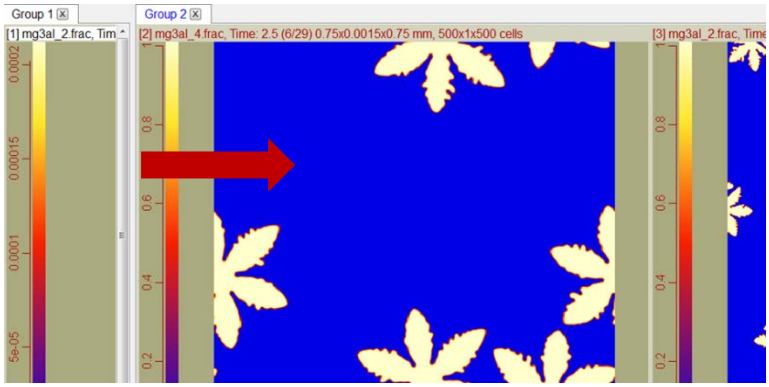


Figure 4.31 Changing the view of two side by side arranged grous

Important is the function of dragging and dropping a dataset from one group to another. There is no selection field for this tool but instead a dataset can be slided to the group where required. This action is accomplished when during the dragging a text box saying '**Move to group**' appears.

4.2.10 Scripting

In Display MICRESS 7 it is possible to automate every user action with scripting. This means that every action the user does with the mouse or keyboard is reproducible with a script without human interaction. The user is able to write a script within Display MICRESS or in any text editor. Additionally, it is possible to automatically generate scripts from the user's actions within the program; this process is called recording. Scripts can be executed from the internal script editor via hotkeys or at the startup of Display MICRESS.

4.2.10.1 Writing scripts

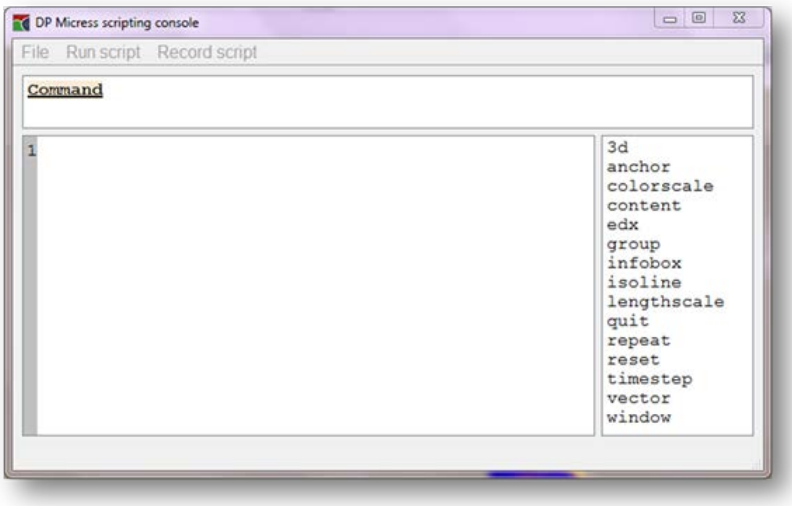


Figure 4.32 The scripting console

As script files are plain text files they can be written in any text editor. However, it is recommended to use the internal editor within Display MICRESS to benefit from syntax highlighting and command references. The editor, called scripting console, can be invoked via "Show scripting console" in the "Scripting" menu.

The editor consists of three sections. The lower left section is the editor area where the user is able to write text. The area above the editor area, called information area, is used to display information about the current command. Every command is defined in a single line and consists of a main command, a sub command and command arguments. While the user enters his commands, the information area will display what the user

has to enter to get a valid command. In practice this means that as soon as the command and sub command have been entered, a list of expected arguments will be shown. Some arguments are optional, they will be written in normal text style, whereas required arguments will be written in bold. The yellow background shows in which command argument the text cursor is currently located. To the right side of the editor area the choice area is located. In this area all possible keyword which the user can currently write are displayed. These keywords can be inserted into the text by double clicking them or by pressing the tab key during editing in the editor area. If multiple keywords are available while pressing the tab key, only the leading characters which are equal in all possible keywords will be inserted. All valid keywords are written in lower cases. If a keyword begins with an upper case character, this indicates that the user has to replace this keyword by its meaning. An example would be “Integer” where the user has to enter an integer number.

Commands are separated by newlines and command arguments are separated by spaces. If an argument needs to contain spaces it can be enclosed by quotation marks. If the first character in a line is ‘#’, the line is considered to be a comment and is thus ignored during execution.

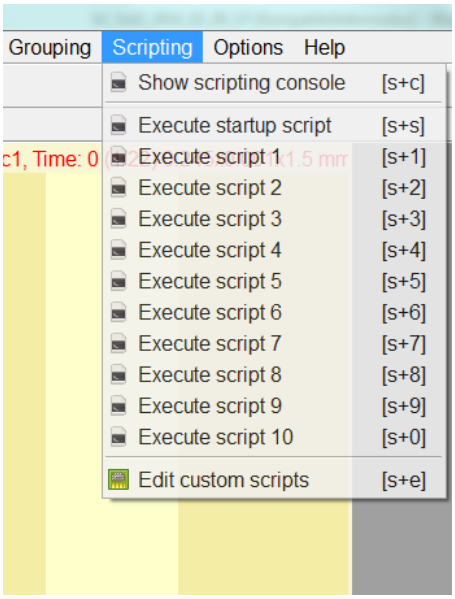
For easier reading of the command lines they are highlighted according to their syntax. Valid commands will be written in blue, valid arguments in green and invalid text will be written in black. If a command line is invalid its background color will be red, valid commands will have a white background color.

As soon as a command line is valid, it can be executed. There are four ways of execution:

- Execute current line, meaning the line where the text curser is located
- Execute all lines from top to bottom
- Execute all lines from the top to the current line, including the current line
- Execute all lines from the current line to the bottom, including the current line

The executions can be inherited via the “Run script” menu or via hotkeys (see “run script” menu).

During execution, the background color in the line number area (left to the editor area) will indicate a line’s execution status. Gray lines are pending, blue lines are currently executing, green lines have been executed successfully and red lines have executed with errors. If a line executed with errors the execution of all other lines will be stopped. Lines will be executed one after another and it is guaranteed that the previous command’s result is available to the next command. For example a newly opened file can be safely referenced to in the next command line.



The ‘Scripting’ menu includes several slots to hold scipts and execute it, especially a startup script slot which will be executed at each DP_MICRESS startup (see figure 4.33). These slots and the containing scripts can be modified by ‘Edit custom scripts’. Do not save these dedicated scripts with another file name because otherwise the connection to the execution slots will be lost (see also section 4.2.10.3).

Figure 4.33 Executing several scripts and ‘Edit custom scripts’

4.2.10.2 Recording scripts

It can be very helpful to not write every single command of a longer command chain, but just let the scripting console record what the user does. To enable this feature recording has to be started via the “Record script” menu in the scripting console. Once recording was started the scripting console will be appended with command lines as soon as the user performs an action within Display MICRESS. Generated scripts should always be read and corrected prior to using them, because some generated command lines can be needless. For example if three EDX points are placed with the mouse, three commands will be generated where only the last one would be needed. Some of the actions are not plotted, for example the line scan export as a table.

After recording the script, be sure to stop recording via the “Record script” menu. Running a script during recording can potentially produce infinite loops because the executed commands will be added to the script again.

4.2.10.3 Default scripts, startup script and command line use

One way of executing a script is to load it in the scripting console and execute it. For scripts which are used very often there is a more convenient method: Custom scripts. There are ten custom scripts which can be executed via the Display MICRESS “Scripting” menu or by a simple hotkey combination. To edit these scripts use “Edit custom scripts” in the “Scripting” menu and select one of the custom scripts. This essentially opens a scripting console with an opened file which can be edited and saved. Please don’t use the “Save as” feature, because this would save the custom script as a new and unrelated script. The custom scripts will automatically be saved in the user’s home directory and are thus personalized for every user. The first line of the script file contains a comment where the user can enter a name for the script; this name will appear in the “Scripting” menu.

There is an eleventh custom scripted called “Startup script”. The startup script behaves like any other custom script with the addition that it will be run automatically after every startup of Display MICRESS 2. If the user passed filenames to Display MICRESS at startup, it is guaranteed that the files will be loaded before the startup script is executed.

The last two ways of running a script file are command line arguments. When starting Display MICRESS 2 the user can add the keyword --script, followed by the filename of a script file. This will result in the script being loaded and run after all data files have been loaded and after the startup script was executed. The --script keyword cannot be followed by multiple script files but it is possible to use the keyword multiple times. If this is done the scripts will be executed in the order in which they were specified. The second command line argument for script execution is --console. This argument behaves like --script, with the exception that the graphical user interface will be created but will stay minimized and thus in the background. It is recommended to end scripts with the “quit” command if they are invoked with --console, as this will create a behavior where Display MICRESS 2 starts up, executes a script and then closes itself without any user interference.

4.2.10.4 Script language definition

Most of the scripting commands consist of a main command, a sub command and some arguments. All performable actions can be grouped in multiple groups which are defined by the main command, meaning that actions which are related will share a main command. This grouping was adapted for the following sub chapters. There exist two commands, which don’t need any sub command or argument:

Command	Explanation
reset	Reset the application as if it was closed and opened again. All open files and groups will be closed.
quit	Quit the application. All unsaved data will be lost.

Some arguments are optional. All optional arguments can be replaced by the '*' sign, which means that the default value of the argument will be used. If a command ends with optional arguments the trailing '*' signs can be omitted.

Some arguments expect a list of integer numbers. This list must contain comma separated numbers without spaces. Dashes can be used to express series of numbers. An example list is "1,2,3-5" meaning 1, 2, 3, 4 and 5.

- Window commands

All following sub commands need to be prefixed with the "window" command and apply on all active windows of the current group, if not specified differently. Some sub commands require window IDs. These can be specified directly by referencing the number shown at the beginning of the header of a window. Another way is to use negative IDs to reference the newest windows. For example -1 would be the newest window, -2 the second newest window, and so on. A window is newer than another if its ID is larger. The same holds for group IDs.

Sub command	Explanation	Arguments
add	Open a dataset and display it in a new window. If one dataset is opened, it will be added to the current group, for multiple datasets a new group will be created. The file name argument can be repeated to open multiple files.	File name: A relative or absolute path to a dataset file
tile	Tile windows. With arguments it is possible to sort the windows according to their windows ID. If not all windows IDs are given, the rest will be sorted from low to high numbers.	Sorted, optional: true or false, to enable or disable sorting. Default is false. Window ID, optional: Specify which windows should be at the first position. This argument can be repeated to specify the ID for the second, third, … position.
cascade	Cascade windows	
fit	Fit the windows sizes to their contents, reducing the amount of unused space.	
mainfit	Fit the main windows to the current group's content such that all windows are visible and no empty space is visible.	
size	Set window sizes to a specific size in pixel. The specified size will apply on the rendered dataset, meaning that with annotations, the actual window size will be larger.	Width (pixel) Height (pixel)
move	Move a window to a certain position inside the group. The position is relative to the upper left corner of the group.	Window ID Horizontal position (pixel) Vertical position (pixel)
arrange	Specify how "tile" should arrange the windows. It is possible to specify a fixed row count, a fixed column count or use automatic mode. Count has to be specified for "rows"	Mode: auto, rows or columns Count, optional

	and “columns”, for “auto” the count will be ignored.	
active	Set active state of windows.	Window ID, optional: If specified only one window will be affected, otherwise (‘*’) all windows of the current group will be affected Active: true or false, default is true
others	Set active state of all windows of the current group but the specified one.	Window ID Active: true or false, default is true
group	Move a window to a specific group.	Window ID Group ID
duplicate	Duplicate a window, creating an exact but independent copy. If no window ID is given, all active windows of the current group will be duplicated.	Window ID, optional
name	Rename a window.	Window ID Window name
remove	Close a window. All unsaved changes will be lost. The window ID is non-optional. To close all windows of the current group use “group clear”.	Window ID

- Group commands

Sub com-mand	Explanation	Arguments
add	Create and show an empty group.	
show	Show a group	Group ID
remove	Close a group, including all of the windows in the group.	Group ID, optional: default is the current group
clear	Close all windows in a group.	Group ID, optional: default is the current group
export	Generate an image from the current group’s active windows.	File name, optional: The default file name depends on the dataset in the first window. The file name’s extension specifies the file format. Time step, optional: Can be “cur”, “all” or a custom list. Default is the current time step. 3rd dimension depth, optional: Can be “cur”, “all” or a custom list. Default is the current depth. Resolution (%), optional: Scales the image to use a finer resolution while showing the same content. 200 means double width and height and thus four times as many pixels. Foreground color, optional: Color for text and lines, format must be a

		hex code like 0xRRGGBB, default is what was last used in an export from within the program. Background color, optional: see foreground color

- Time step commands

All following sub commands need to be prefixed with the “timestep” command. All commands are applied on all active windows of the current group.

Sub command	Explanation	Arguments
next	Show next time step	
previous	Show previous time step	
first	Show first time step	
last	Show last time step	
set	Set time step to a specific number. If the number is larger than the amount of available time steps, the last available time step will be shown.	Time step number

- 3rd dimension commands

All following sub commands need to be prefixed with the “3d” command. All commands are applied on all active windows of the current group.

Sub command	Explanation	Arguments
plane	Define which two axes should be displayed and thus which dimension should be constant.	Orientation: “xy”,”xz” or “yz”
next	Move the constant dimension one cell up	
previous	Move the constant dimension one cell down	
set	Set the constant dimension to a specific number. If the number is larger than the size of the dimension, the last available depth will be shown.	Depth (cells)

- Repetition and mirroring commands

All following sub commands need to be prefixed with the “repeat” command. All commands are applied on all active windows of the current group.

Sub command	Explanation	Arguments
horizontal	Set amount of horizontal repetitions with or without mirroring, allowed values are 1, 2 and 3.	Amount Mirrored, optional: true or false, default is false

vertical	Set amount of vertical repetitions with or without mirroring, allowed values are 1, 2 and 3.	Amount Mirrored, optional: true or false, default is false

Content commands

All following sub commands need to be prefixed with the “content” command. Most commands are applied on all active windows of the current group if the window ID is optional and ignored. The window ID argument follows the same rules as described in chapter X.4.1.

Sub command	Explanation	Arguments
zoom	Zoom to a certain area by specifying its borders (including). If one of the border values is left blank, the current value will be kept.	Window ID, optional Horizontal minimum (cells), optional Horizontal maximum (cells), optional Vertical minimum (cells), optional Vertical maximum (cells), optional Hide outside cells, optional: true or false, default is false
ratio	Modify pixel ratio. The ratio (pixel width)/(pixel height) will be (width ratio)/(height ratio).	Window ID, optional Width ratio Height ratio
info	Show statistical properties of a window on the console or store it in a file. The properties can be calculated for the current region or the complete dataset, and for the current time step (Mode single), one value for all time steps (Mode complete) or as one value for every time step (Mode list).	Window ID Region, optional: current or global, default is current Mode, optional: One of single, complete, list, default is single File name, optional: If non name is given the output will be written to the console
operation	Set operation of a window. If no dataset ID is specified, the old datasets will be kept. The pair of dataset ID and time step modification can be repeated up to 22 times to define multiple datasets as variables a to w excluding t.	Window ID Operation Dataset ID, optional Time step modification, optional
grid	Set comparison grid mode for operations to cell based or scale based.	Window ID, optional Grid mode: cell or scale
subpixel	Enable or disable sub pixel rendering.	Window ID, optional Sub pixel rendering: true or false
compare	Create a new window and compare two datasets in it. First dataset will be a, second will be b.	First dataset ID Second dataset ID Operation, optional: Default is a-b Time step modification, optional: Default is

		0
export	Export windows content as dataset file. It is possible to export the current time step, all time steps or a list of time steps. The file format can be the MICRESS geoF based data format or a binary or ASCII vtk file. For vtk files it is possible to embed multiple datasets in one file. Additional datasets can be specified by adding one or multiple pairs of additional window IDs and data types. The output files can be compressed; whether or not compression is enabled by default depends on the chosen file format. If geometry values are defined, the dataset will be transformed to fit the new geometry. The default transformation algorithm is nearest neighbor but a trilinear algorithm can be enabled. For more information about the algorithms see chapter <muss noch geschrieben werden, vermutlich 4.2.?>.	Window ID File name, optional: Default is based on first dataset in window Time step, optional: cur, all or a list, default is all File format, optional: One of geof, vtk, vtkascii, default is geof Compress file, optional: true or false, default is true for geof and false for both vtks Data type, optional: Default is data type of first dataset in window Current region, optional: true or false, default is false Geometry X, optional Geometry Y, optional Geometry Z, optional Trilinear, optional: true or false, default is false Additional window ID, optional Additional datatype, optional

- Color scale commands

All following sub commands need to be prefixed with the “colorscale” command. Most commands are applied on all active windows of the current group. The window and dataset ID arguments follow the same rules as described in 4.2.5.1 .

Sub command	Explanation	Arguments
show	Show or hide color scale.	Visible: true or false
color	Set color scheme to one of the predefined schemes. Valid values are integer values from 1 to 11.	Color scheme
load	Load and use a color scheme from a file.	File name
position	Set the color scale to a certain position.	Position: One of left, right, top, bottom, independent
flip	Flip color scale.	
gap	Set color scale gap.	Gap (pixel)
size	Set color scale size.	Size (pixel)
range	Set color scale range to a fixed minimum and maximum. This command can be	Window ID, optional: Default is all active windows

	applied to all or one color scale(s). No minimum or maximum value will enable automatic mode. Mixing automatic mode and fixed values is possible. The minimum and maximum values will be calculated from the specified region.	Minimum value, optional: Default is automatic mode Maximum value, optional: Default is automatic mode Region, optional: One of cur, timestep or all, default is cur
rectangle	Set color scale range to a fixed minimum and maximum based on the values in a specific region. If one of the rectangles coordinates is left blank it will be set to one (min) or the maximum possible value (max).	Window ID, optional: Default is all active windows Horizontal minimum (cells), optional Horizontal maximum (cells), optional Vertical minimum (cells), optional Vertical maximum (cells), optional

- Length scale commands

All following sub commands need to be prefixed with the “lengthscale” command. All commands are applied on all active windows of the current group.

Sub command	Explanation	Arguments
show	Show or hide length scale.	Visible: true or false
position	Set the length scale to a certain position.	Position: One of horizontal, vertical
size	Set length scale size.	Size (pixel)
length	Set length scale length. No length means automatic mode. Length is an integer value in micrometer or nanometer, depending on Nanometer.	Length, optional Nanometer, optional: true or false, default is false

- Information box commands

All following sub commands need to be prefixed with the “infobox” command. All commands are applied on all active windows of the current group.

Sub command	Explanation	Arguments
show	Show or hide information box.	Visible: true or false
position	Set the information box to a certain position.	Position: One of left, right, top, bottom
size	Set information box size.	Size (pixel)
content	Set information box text content	Content: String
icon	Show or hide MICRESS icon.	Show: true or false

- Vector commands

All following sub commands need to be prefixed with the “vector” command. Most commands are applied on all active windows of the current group. The window and dataset ID arguments follows the same rules as described in chapter X.4.1.

Sub command	Explanation	Arguments
show	Create a new window and show a vector dataset.	Background window ID Horizontal dataset ID Vertical dataset ID
grid	Set vector grid size.	Grid size (pixel)
cutoff	Set threshold for minimum and maximum vector length. No value means no threshold.	Minimum value, optional Maximum value, optional
type	Set visualization type to arrows or lines.	Visualization: One of arrows, lines
color	Set visualization color.	Color: Hex code like 0xRRGGBB

- Isoline commands

All following sub commands need to be prefixed with the “isoline” command. Most commands are applied on all active windows of the current group. The window and dataset ID arguments follows the same rules as described in chapter 4.4.1.

Sub command	Explanation	Arguments
set	Set values for isoline generation.	Value(s): Infinite value arguments possible
norm	Set values for isoline generation to n uniformly distributed values reaching from the minimum to the maximum values of the current region.	Value count
color	Set color for isoline visualization. It is possible to specify multiple color pairs, up to one for every isoline value. If fewer colors are specified, the first lines will have custom colors and the last lines will have the color of the last specified color.	Inner color: Hex code like 0xRRGGBB Outer color: Hex code like 0xRRGGBB
width	Set the width of the isoline visualization.	Width (pixel)
timestep	Enable or disable preservation of isolines of different time steps on time step change.	Preserve: true or false
transparency	Set background transparency.	Transparency (%)
source	Define which dataset should be used for isoline generation in a specific window. If no dataset is specified the visible content (including operations) will be used for gen-	Window ID Dataset ID, optional

	eration.	
export	Save isoline contour values in an ASCII file. The export can be done for all or the current time step and for the current region or the complete dataset geometry.	- All time steps: true or false - Complete dataset: true or false - File name

- EDX commands

All following sub commands need to be prefixed with the “edx” command. Most commands are applied on all active windows of the current group. The window and dataset ID arguments follows the same rules as described in chapter X.4.1.

Sub command	Explanation	Arguments
points	Set X/Y/Z coordinates of EDX points. It is possible to specify an infinite amount of coordinate triples. Invoking this command will delete all previously existing EDX points.	Window ID, optional: No ID means all active windows X Coordinate (cell) Y Coordinate (cell) Z Coordinate (cell)
cells	Set cell pointer coordinates of EDX points. It is possible to specify an infinite amount of cell pointers. Invoking this command will delete all previously existing EDX points.	Window ID, optional: No ID means all active windows Cell pointer
line	Select a default EDX line. Invoking this command will delete all previously existing EDX points.	Position: One of left, right, top, bottom, horizontal, vertical, ascending, descending
show	Show EDX analysis in a graph window. The five available modes are “Show line data” (line), “Show line data over time” (linetime), “Show point data over time” (point), “Show value on line over time” (value) and “Show non-zero values on line over time” (nonzero). The time step values can be adjusted in the script, all other values have to be adjusted in the graph window.	Mode: One of line, linetime, point, value Time step (from), optional: Integer number or cur for current, default is cur Time step (to), optional: Integer number or cur for current, default is cur
export	Export EDX analysis values as table. For available modes see “edx show”. The time step values can be adjusted. For the mode “value”, the value can be set with the Search value attribute.	Mode: One of line, linetime, point, value Time step (from), optional: Integer number or cur for current, default is cur Time step (to), optional: Integer number or cur for current, default is cur Filter value, optinal: If a x-value equals this

		value, the corresponding cell will not be included in the export. Search value, optional

- Anchor commands

All following sub commands need to be prefixed with the “anchor” command. The window and anchor ID arguments follows the same rules as described in chapter X.4.1.

Sub command	Explanation	Arguments
add	Add new anchor to a window. If no label coordinates are given the label will be placed close to the head.	Window ID Head X Head Y Label X, optional Label Y, optional
move	Move an anchor and its label. If values are unspecified they will not be changed.	Anchor ID Head X, optional Head Y, optional Label X, optional Label Y, optional
text	Set an anchor’s text.	Anchor ID Text
head	Specify an anchor’s head appearance.	Anchor ID Head: One of arrow, point, plain
foreground	Set foreground color (text and lines) and opacity. If only one value is specified the other will stay unmodified. Opacity doesn’t influence the text.	Anchor ID Color, optional: Hex code like 0xRRGGBB Opacity (%), optional
background	Set background color and opacity. If only one value is specified the other will stay unmodified.	Anchor ID Color, optional: Hex code like 0xRRGGBB Opacity (%), optional
arrow	Show or hide anchor arrow including arrow head.	Anchor ID Visible: true or false
border	Show or hide anchor border.	Anchor ID Visible: true or false
remove	Remove an anchor.	Anchor ID

4.2.11 Options

In the three superordinated categories ‘Appearance’, ‘Memory’, and ‘Miscellaneous’ there are several basic settings selectable. Firstly, in ‘Appearance’ the font, its size, and the window colors of the DP_MICRESS layout can be chosen.

Secondly, ‘Memory’ contains important features providing a smooth operation (See Figure 4.34).

‘Maximum memory usage for timestep caching’ limits the amount of memory which will be used for caching time steps in main memory which are shown before. Raising this limit will smooth the stepping through time steps, esp. for large datasets.

‘Don’t cache timesteps which are larger than (% of cache memory)’ This parameter allows to control the granularity of the time step cache. It depends on the size of the data which percentage of cache memory is a useful setting. The pretty small default value is chosen to hinder few large time steps taking the whole cache memory.

‘Temporarily uncompress files which are larger than (MB)’ For large compressed data files it is more efficient to uncompress them once before showing them. All following read operations will access the uncompressed file and will be faster. However, disk space for the set temporary directory might be limiting.

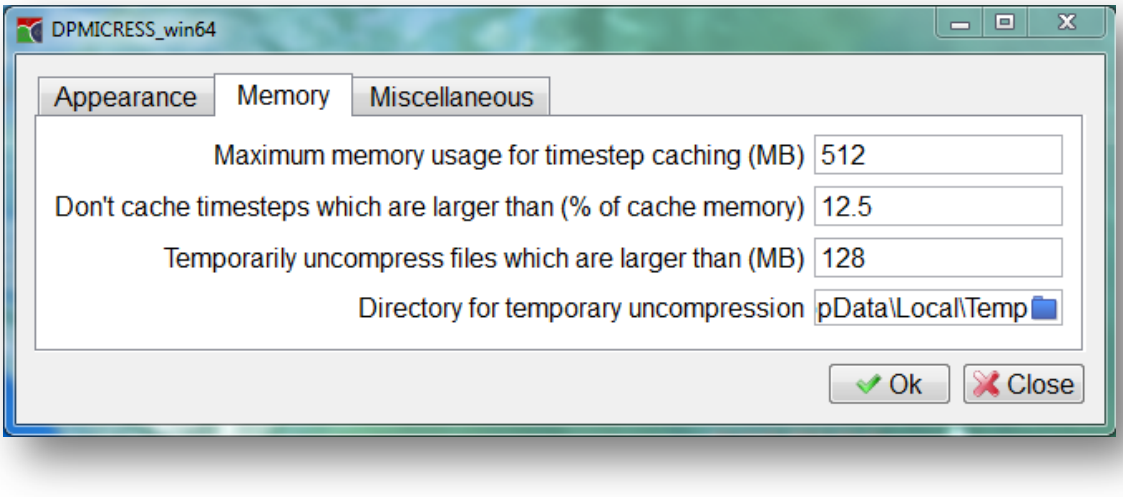


Figure 4.34 The DP-MICRESS memory settings

Thirdly, the `Miscellaneous’ settings contain the option to hide confirmation requests such as `Do you really want to exit Display Micress?’. This rubric is also handy in terms of hotkeys, presented in chapter 4.3. Once a part of a hotkey is entered it can be complemented for 2 seconds. The `Time between frames in animation (seconds)’ refers on the possibility to make a video out of the simulation. When filming the on-going simulation the time space between two simulation steps is fixed. So the video seems more fluent or rather slowly for a detailed observing (See Figure 4.35).

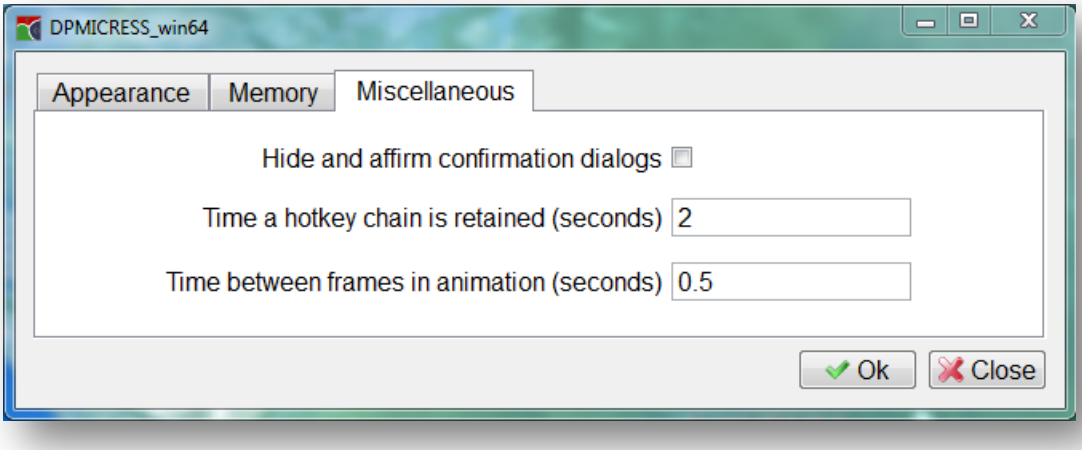


Figure 4.35 Useful settings

4.3 Hotkey chains

As presented in Figure 4.36d, DP_MICRESS® always informs the user about the currently possible shortcuts to modificate an active result file. The shortcuts are organized in way that they are combined out of a sum of the first letter of the superordinated window rubric plus another fitting letter. Since there are collosions in `Animation' and `Annotations', `Annotations' is supplied with another nomenclature. Correspondingly Table 4.3. sums up all the available shortcuts.

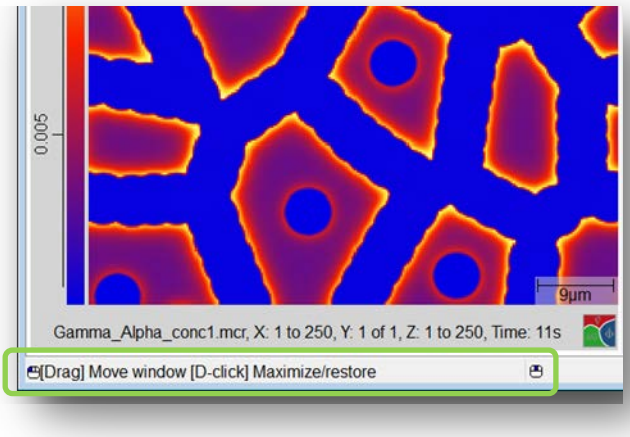


Figure 4.36 The usable shortcuts for further working on the active result file

Table 4.3 The shortcuts being available for DP_MICRESS®

Rubric	Function	Hotkey
File		
	Open solution	f+o
	Reload all solutions	f+r
	Export window contents as dataset	f+d
	Export group as image	f+i
	Quit	f+q
Window		
	Tile arrangement	w+a
	Tile windows	w+t
	Cascade windows	w+c
	Set window sizes	w+s
	Fit window sizes	w+f
	Fit main window size	w+m
	Duplicate all windows	w+d
	Close all windows	w+x
Content		
3D	XY cutting plane	3+z
	XZ cutting plane	3+y
	YZ cutting plane	3+x
	Cutting plane up	3+u

Repetition and Mirroring	Cutting plane down	3+d
	horizontal, disabled	h+d
	horizontal, repeat 2x	h+r+2
	horizontal, repeat 3x	h+r+3
	horizontal, mirror 2x	h+m+2
	horizontal, mirror 3x	h+m+3
	vertical, disabled	v+d
	vertical, repeat 2x	v+r+2
	vertical, repeat 3x	v+r+3
	vertical, mirror 2x	v+m+2
	vertical, mirror 3x	v+m+3
	Zoom	d+z
	Properties	d+p
	Ratio	d+r
	Operations	d+o
	Show vectors	d+v
	Vector settings	d+s
	Isoline settings	d+i
Annotations		
Color Scale	Show/hide annotation	c+a
	coloring 01	c+0+1

	coloring 02	c+0+2
	coloring 03	c+0+3
	coloring 04	c+0+4
	coloring 05	c+0+5
	coloring 06	c+0+6
	coloring 07	c+0+7
	coloring 08	c+0+8
	coloring 09	c+0+9
	coloring 10	c+1+0
	coloring 11	c+1+1
	Position- Right	c+p+r
	Position-Top	c+p+t
	Position-Left	c+p+l
	Position-Bottom	c+p+b
	Independent	c+i
	Flip scale	c+f
	Set gap	c+g
	Set range	c+r
	Set size	c+s
Length Scale	Show/hide annotation	l+a
	Horizontal	l+h

Information box	Vertical	l+v
	Set size	l+s
	Show/hide annotation	i+a
	Right	i+r
	Top	i+t
	Left	i+l
	Bottom	i+b
	Set content	i+c
	Set size	i+s
	Show/hide icon	i+w
Animation		
	Start/stop animation	a+s
	First step	a+f
	Previous step	a+p
	Next step	a+n
	Last step	a+l
Virtual EDX		
Pick default line	Horizontal	e+d+h
	Vertical	e+d+v
	Right	e+d+r
	Top	e+d+t

	Left	e+d+l
	Bottom	e+d+b
	Ascending diagonal	e+d+a
	Descending diagonal	e+d+d
	Pick points/lines with mouse	e+m
	Pick points/lines with coordinates	e+c
	Show line data	e+l
	Show line data over time	e+t
	Show point data over time	e+p
	Show value on line over time	e+v
	Show non-zero values on line over time	e+n
Mouse pointer		
	Zoom and move	m+z
	Color scale min/max rectangle	m+c
	Points and lines (Virtual EDX)	m+p
	Anchor label	m+a
Grouping		
	Create new group	g+c
Side by side direction	horizontal	g+h
	vertical	g+v
Scripting		

	Show scripting console	s+c
	Execute startup script	s+s
	Execute script 1	s+1
	Execute script 2	s+2
	Execute script 3	s+3
	Execute script 4	s+4
	Execute script 5	s+5
	Execute script 6	s+6
	Execute script 7	s+7
	Execute script 8	s+8
	Execute script 9	s+9
	Execute script 10	s+0
	Edit custom scripts	s+e
Options		
	Edit settings	o+e
	Restore default settings	o+r
Help		
	Display Micress Handbook	F1
	About Display Micress	F2

5 ParaView

5.1 Installing and running ParaView

The freeware ParaView is especially useful for post-processing of 3D MICRESS output data. The software can be downloaded for free from www.paraview.org.

ParaView requires a vtk-format of the files to be read. To exploit how to create a vtk-file from existing MICRESS results, please refer to chapter 7 of this handbook on "Data conversion". There are also options to directly specify/create vtk output from MICRESS simulations. Please note, that within this manual only a very basic introduction can be given, as there are many, many options for the use of ParaView. To open an existing vtk-file in ParaView, run the program. In the main menu bar, go to File → Open. Choose the file to be opened. It will usually appear as "filename_.vtk". The selected file appears in the tree of built-in objects in the left side of the ParaView window. Click on "apply" under object inspector → Properties (the sub window below the built-in objects tree). Then, the calculation domain (similar to Figure 5.1) of your simulation will appear in the window on the right.

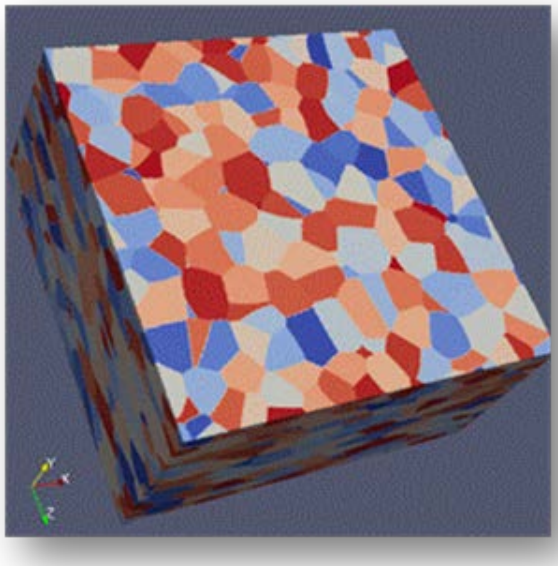


Figure 5.1 A MICRESS 3D result output displayed using ParaView

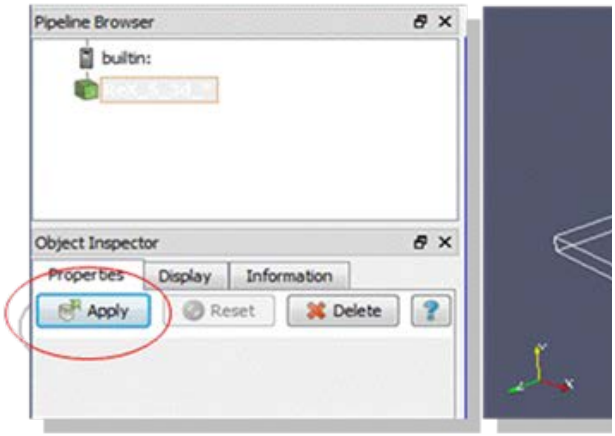


Figure 5.2 Opening a vtk file in ParaView (left) and the simulation domain (right)

5.2 Using ParaView

5.2.1 Display surface or volume

Once having loaded the result file, use the button shown in fig. 5.3 (marked in red) to switch among „outline“, „surface“, „volume“ and „slice“. Observe how the simulation domain changes its appearance.

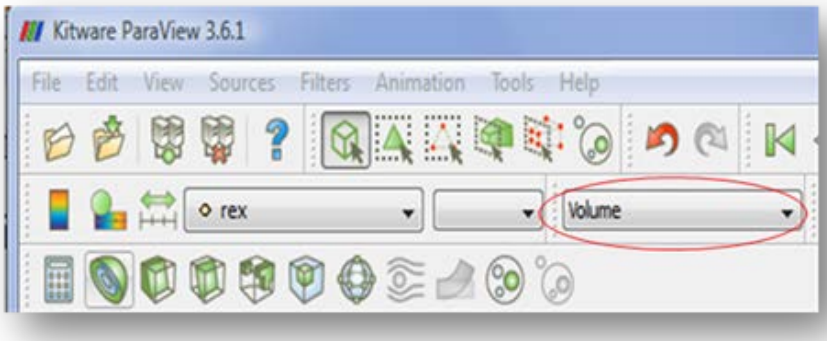


Figure 5.3 The ParaView main menu bar: the Volume feature

5.2.2 Rotation/ Translation

Pressing the left mouse button and moving the mouse at the same tile leads to free rotation. Pressing the right mouse button with moving the mouse simultaneously zooms in and out. A rotation around an axis is possible via pressing the „shift“ and the „left“ mouse button with simultaneously moving the mouse. For translation, press the „shift“ button and the „right“ mouse one and move the mouse.

5.2.3 Colour by different results and attributes

Use the button coloured in red (Figure 5.4.) to switch between different output results. The simulation domain will be coloured according to the output file selected in the menu bar (see Figure 5.5)

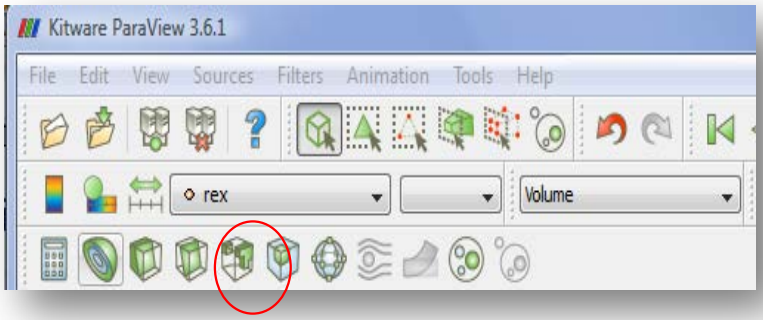


Figure 5.4 The ParaView main menu bar: the attribute feature

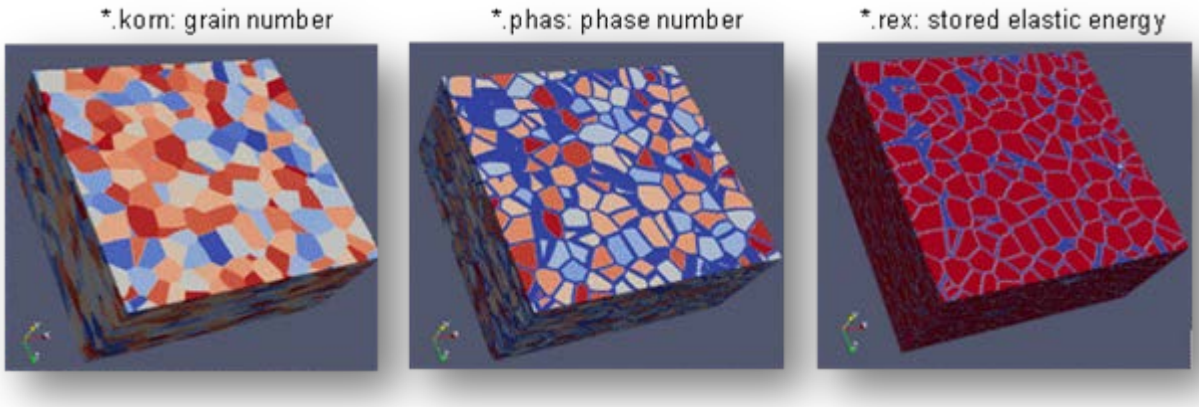


Figure 5.5 Colouring the simulation domain according to the different attributes of the output file

5.2.4 Run an animation

You can run an animation using the output for a given attribute for all time steps. For that purpose, use the animation bar in the main menu. You can easily switch between previous and next time steps, go directly to the very first/last time step or perform different loops of visualisation.

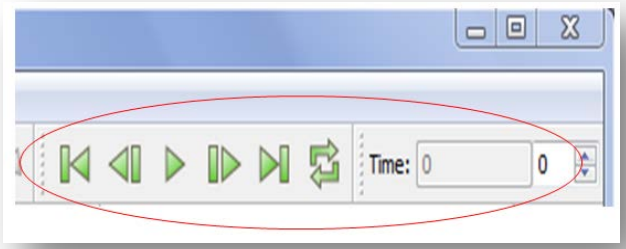


Figure 5.6 The ParaView animation bar

5.2.5 Applying filters/setting thresholds/making sections

The clip filter cuts away a portion of the input data set using an implicit plane. It returns unstructured grid data on output. Some of the possible options here are a slice filter, a contour filter, a clip filter and others. With their help, one can easily generate different sections through the calculation domain.

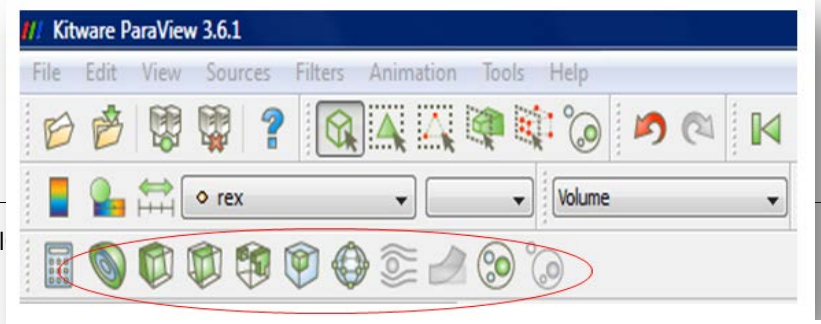


Figure 5.7 The ParaView bar for creating clips and thresholds

More detailed information about that and all other features of ParaView can be found in the ParaView user's manual. Further information is available at www.paraview.org.

6 gnuplot

gnuplot is a command-line driven interactive function plotting utility for both Windows and Linux platforms. The software is distributed freely and can be downloaded from the internet along with tutorials and user-manuals. Gnuplot handles both curves (2D) and surfaces (3D). Surfaces can be plotted as a mesh fitting the specified function, floating in the 3D coordinate space, or as a contour plot in the X-Y plane. For 2D plots, there are also many plot styles including lines, points, boxes, heat maps, stacked histograms and contoured projections of 3D data. The interface includes command-line editing and history in most platforms. Scope of this chapter is only giving a basic introduction to gnuplot using some MICRESS results as examples. For detailed information the reader is referred to the gnuplot manuals.

6.1 Installing gnuplot

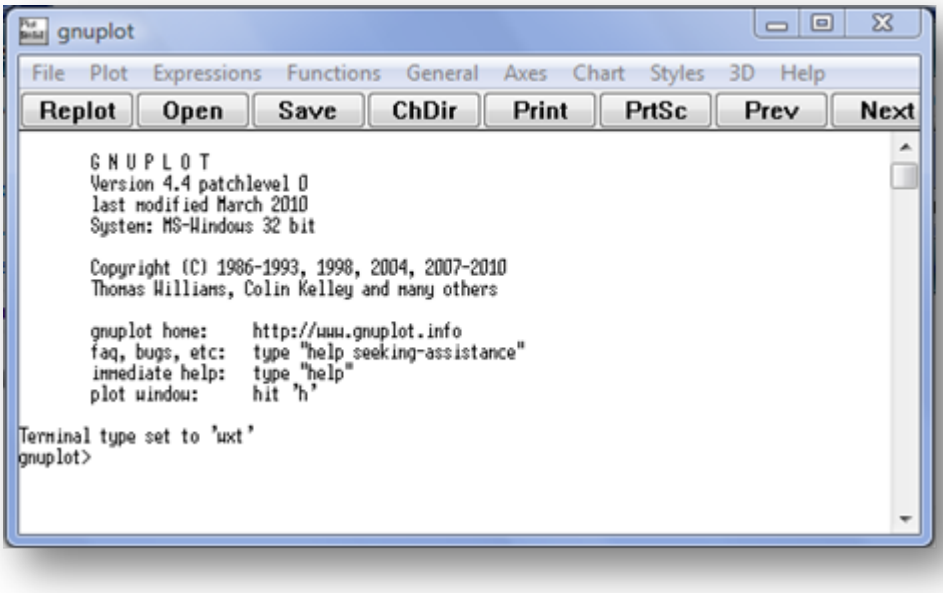

gnuplot
File Plot Expressions Functions General Axes Chart Styles 3D Help
Replot Open Save ChDir Print PrtSc Prev Next
G N U P L O T
Version 4.4 patchlevel 0
last modified March 2010
System: MS-Windows 32 bit
Copyright (C) 1986-1993, 1998, 2004, 2007-2010
Thomas Williams, Colin Kelley and many others
gnuplot home: http://www.gnuplot.info
faq, bugs, etc: type "help seeking-assistance"
immediate help: type "help"
plot window: hit 'h'
Terminal type set to 'wxt'
gnuplot>

Figure 6.1 The gnuplot command line after installing gnuplot

In order to install gnuplot, first download the installation package from the World Wide Web and then unzip it into an appropriate directory. After having read the “Readme” files and installation instructions, you can run the wgnuplot.exe, Fehler! Verweisquelle konnte nicht gefunden werden.see Figure 6.1. Using the command-line, you can proceed to visualize and post-process MICRESS® output files

6.2 Plotting data and fitting curves

The possibilities of plotting MICRESS® data and fitting curves will be explained with the help of the “Grain Growth” example. This example is stored in the MICRESS® installation directory.

Let's consider that you would like to plot the average grain radius over time. This information can be found in the Grain_Growth_.TabK -file (see Figure 6.3.) in the “Grain Growth” result directory. The details for this procedure are explained in Fig. 6.2:

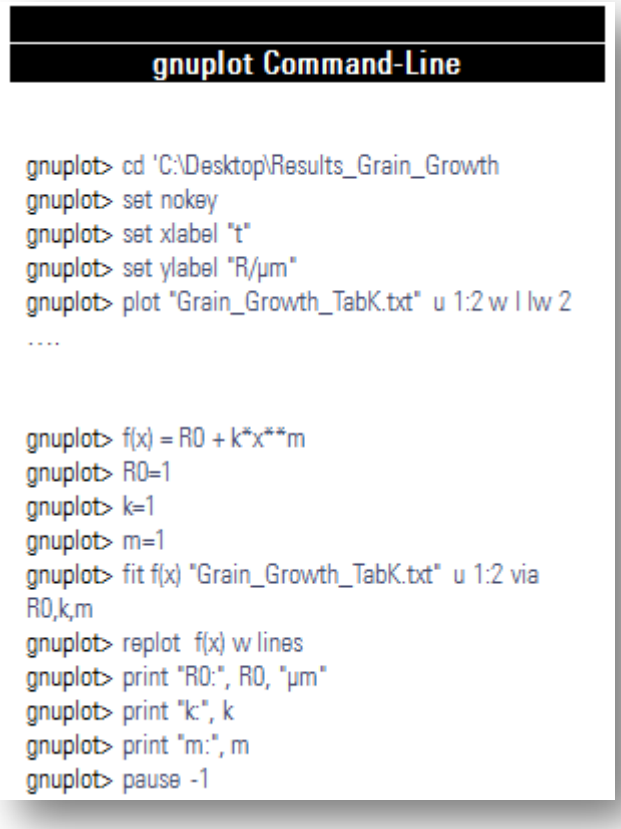
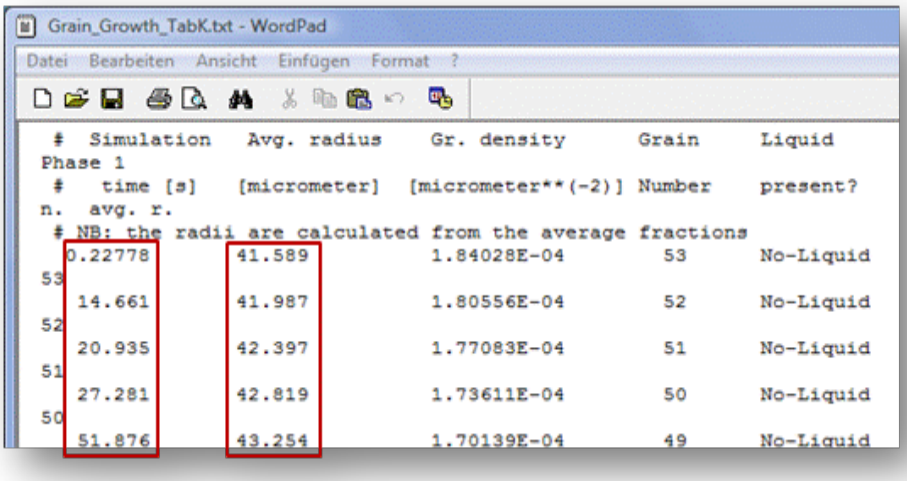


Figure 6.2 Command-line input for plotting data and fitting curves

First specify the path of the respective result file. Then label the axes of the desired plot. If you do not wish to have a legend, use the command “set nokey”. Since this is a 2D plot, only labels for the X and the Y axis are necessary. This is done with the commands set xlabel and set ylabel, resp. The time t for microstructure evolution will be plotted along the X-axis and the grain radius in micrometers will be plotted along the Y-axis. The columns of the data to be plotted are specified by typing e.g. “plot "Grain_Growth_TabK.txt" u 1:2 w l lw 2” where “u 1:2” indicates, that column 2 of the table (see Fig 6.2) shall be plotted over column 1 by using a line (w l “with line”) and the line width (lw) shall be 2.

As soon as the last command is entered the plot shown in Fig 6.4 will appear. If you wish to fit the curve e.g. by a least squares fit, specify an analytical function describing the curve. A function of the form f(x) = R0 + k*x**m has been used as an example here (Fig 6.2, second part of the gnuplot commands).


Grain_Growth_TabK.txt - WordPad

Datei Bearbeiten Ansicht Einfügen Format ?

```
 #  Simulation    Avg. radius     Gr. density       Grain      Liquid
Phase 1
 #    time [s]    [micrometer]   [micrometer**(-2)] Number     present?
n.  avg. r.
 # NB: the radii are calculated from the average fractions
    0.22778         41.589          1.84028E-04        53       No-Liquid
53
    14.661          41.987          1.80556E-04        52       No-Liquid
52
    20.935          42.397          1.77083E-04        51       No-Liquid
51
    27.281          42.819          1.73611E-04        50       No-Liquid
50
    51.876          43.254          1.70139E-04        49       No-Liquid
```

Figure 6.3 The Grain_Growth_TabK.txt-output file: the average radius (column 2) is plotted over the simulation time (column 1)

Initial values for the parameters R0, k and m have to be specified which will be used as initial guesses during the fitting procedure. Type in “fit f(x) "Grain_Growth_TabK.txt" u 1:2 via R0,k,m” to start the fitting procedure. After the optimal fit has been found, the number of iterations that have been necessary for convergence will be shown along with the final sum of squares of the residuals, as well as the fitted values for the parameters R0, k and m. If you wish to plot the value of each of the parameters individually, use the command “print” and type in the parameter and its unit. To plot both the original and the fitted curve, use the command replot as “replot f(x) w lines”. The plot shown in Fig 6.5 will be displayed.

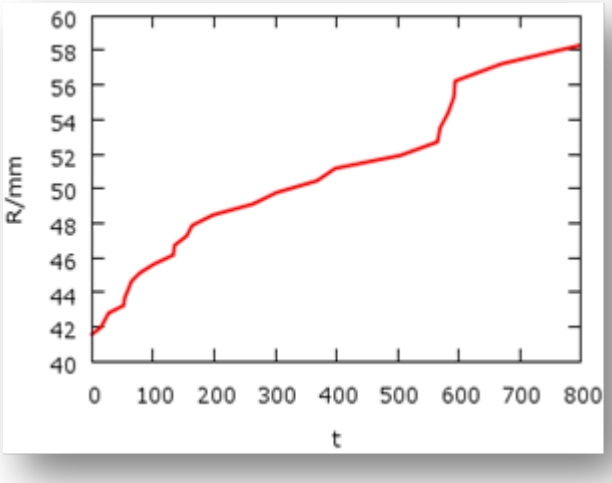


Figure 6.4 A plot of the grain radius over time

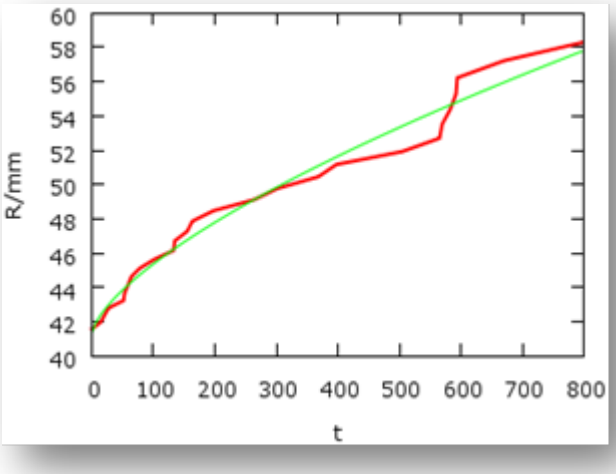


Figure 6.5 The original data (red) and the fitted curve (green)

6.3 Comparing different data

For plotting two different curves in the same figure, use the commands plot and replot. First, set labels for the x and the y axes. Then, use the command “plot” to plot the first curve (see figure 6.6 for syntax). In that way, different curves can be compared to each other (Figure 6.7).

```
gnuplot Command-Line
gnuplot> cd 'C:\Desktop\Results_Gamma_Alpha'
gnuplot> set xlabel "t"
gnuplot> set ylabel "cumulated CPU-time"
gnuplot> plot "Gamma_Alpha_TabP.txt" u 1:2 w l
gnuplot> replot "Gamma_Alpha_TQ_TabP.txt" u 1:2 w l
```

Figure 6.6 Command-line input for plotting and comparing data

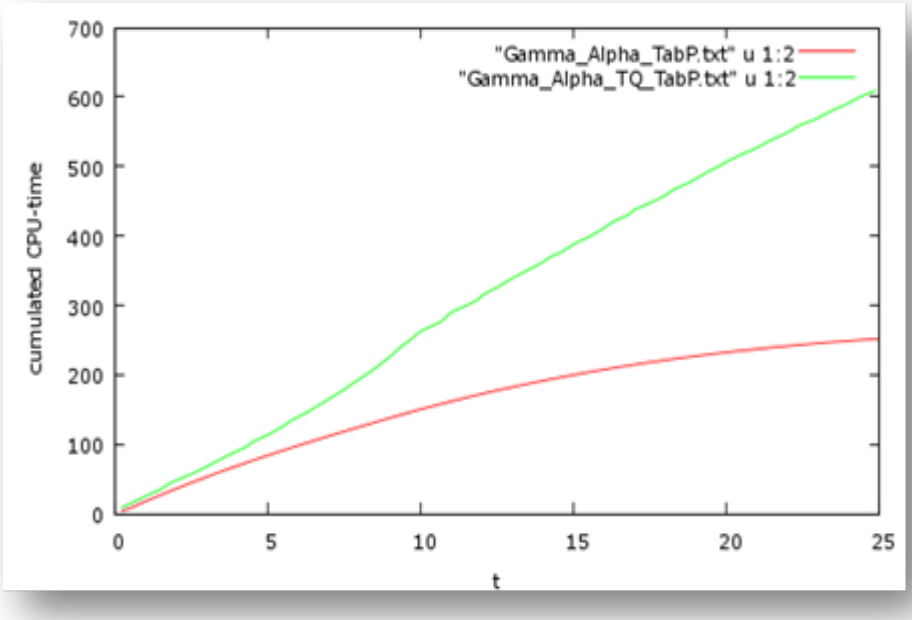


Figure 6.7 Comparing different output files by simultaneous plotting

This chapter has provided some very simple examples of MICRESS® data evaluation using gnuplot. Additional information to exploit all features of gnuplot can be found on www.gnuplot.info.

7 Data conversion: *.mcr to *.vtk

As far as the post-processing of MICRESS® results is concerned, the main characteristic feature of the ParaView software is that it requires a vtk-format of the files to be read. If you want to post-process already existing result files, you need to convert them into a vtk- format which can be done using DP_MICRESS.

For that purpose, start DP_MICRESS, go to 'File' in the main menu bar and from there select the result file to be converted (see Figure 7.2).

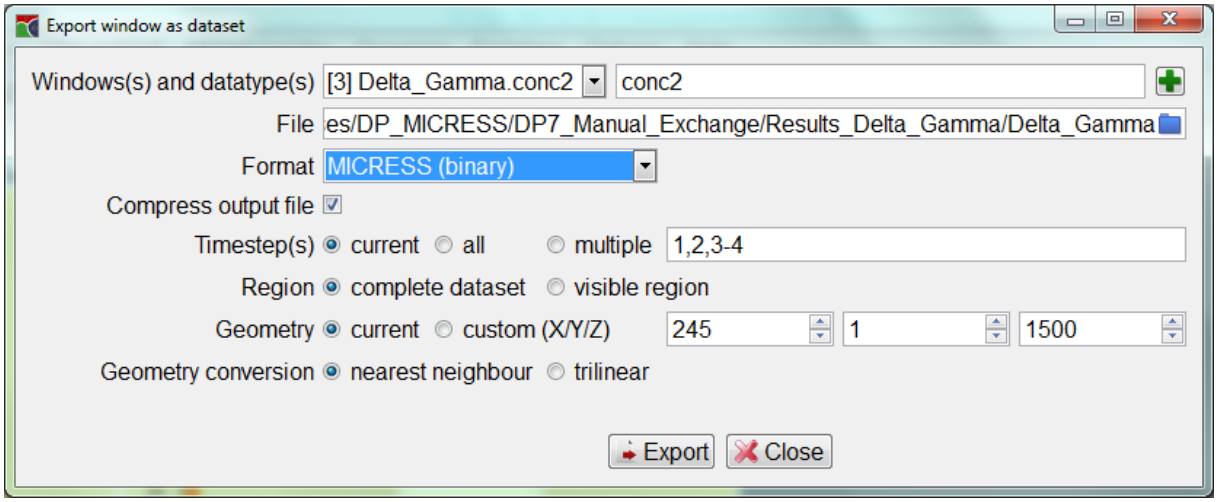


Figure 7.2 The 'Export window as dataset' tool

In the rubric '**Format**' there can be defined in which format the prospective dataset shall be converted. Besides the MICRESS (binary) format, the following options are available: VTK (binary) and VTK (ascii). As Figure 7.3 shows, there are further settings which can be selected for an optimal configurating.

It does not matter which mcr-file you will select. Once you have selected one and given it a respective name to which it shall be converted, click on the '**Export**' option at the bottom of the window.

To get a VTK format from a new MICRESS simulation, choose "overwrite vtk", "binary_zip" in the driving file section "Name of output files". The VTK format results for all field variables, especially 3D, for the output time steps. The time step number will be inserted before the extension ".vtk". To see how to open an existing VTK file in ParaView, please refer to chapter "ParaView".

8 Post Processing: Exercises

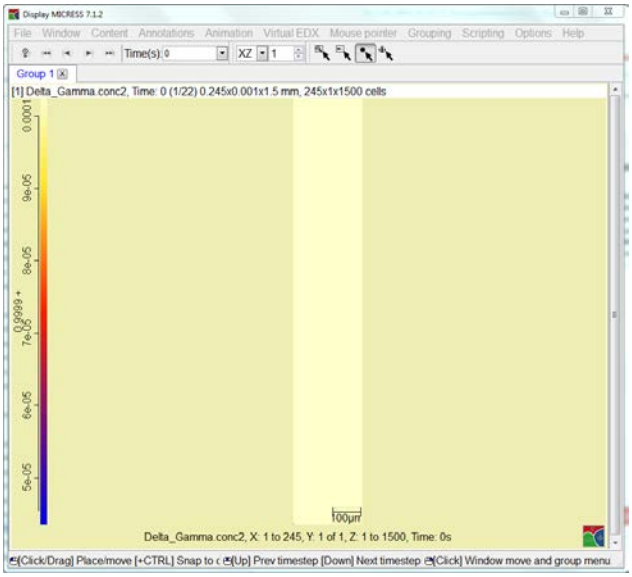
8.1 Introduction to post-processing exercises with DP_MICRESS

This chapter presents a few examples of post-processing using DP_MICRESS. All exercises on post-processing are based on the Delta_Gamma example available in the MICRESS installation directory Not everything simulated here may be meaningful in terms of metallurgy. The same example is however used for didactic reasons with the aim to demonstrate the general approach when post-processing MICRESS results. The given exercises should be performed interactively.

Three exercises will be presented separately within the next sections starting from post-processing of (i) a single result file for a single time step, (ii) a single result file with multiple time steps and (iii) multiple result files simultaneously.

8.2 Read and handle a single result file for a single time step

To begin with, read the result file for conc2 from the Results_Delta_Gamma folder in your MICRESS installation directory.


Group 1
[1] Delta_Gamma.conc2, Time: 0 (1/22) 0.245x0.001x1.5 mm, 245x1x1500 cells
100µm
Delta_Gamma.conc2, X: 1 to 245, Y: 1 of 1, Z: 1 to 1500, Time: 0s

After the data file has been loaded in DP_MICRESS, go to the result at the time of 25 seconds either by using the Next button or the Jump function (See Figure 8.2). Then, remove all text annotations from the figure: View -> Annotations (See Figure 8.3). Next, increase the thickness of your colour scale by going to View -> Increase “thickness” of colour scale (See Figure 8.4). Set three-fold symmetry to make it look like a dendrite. This can be done using the function Display -> Three-fold Symmetry (See Fig. 8.5).

Figure 8.1 The Delta_Gamma.conc2 result file

You can toggle on and off the scale bar with the function View -> Annotations. Click several times on Annotations until you toggle it off. Repeat this action to toggle it on again. Set the scale bar to a typical value (e.g. 100 microns). For that purpose, go to scale -> set range.

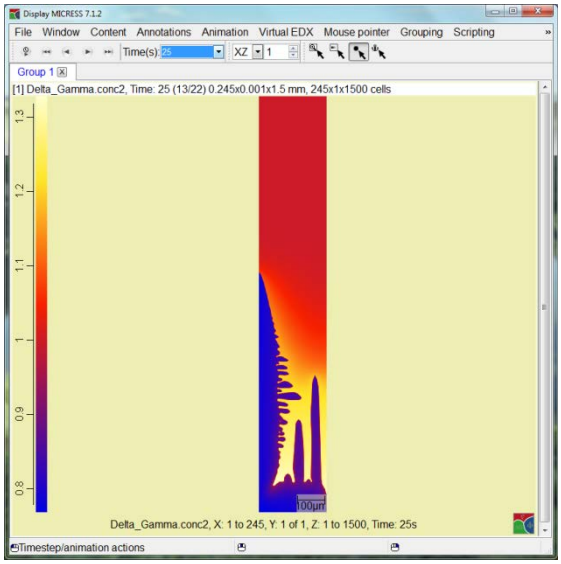

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting
Time(s): 25
XZ 1
Group 1
[1] Delta_Gamma.conc2, Time: 25 (13/22) 0.245x0.001x1.5 mm, 245x1x1500 cells
100µm
Delta_Gamma.conc2, X: 1 to 245, Y: 1 of 1, Z: 1 to 1500, Time: 25s
Timestep/animation actions

Figure 8.2 Moving ahead to time step 25

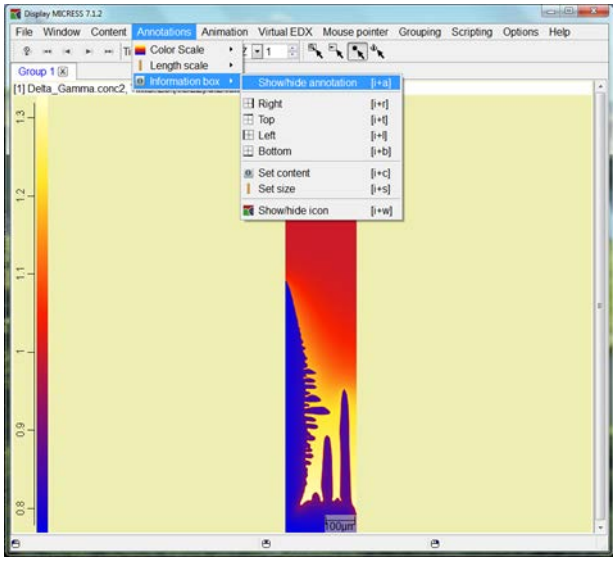


Figure 8.3 Removing the annotations

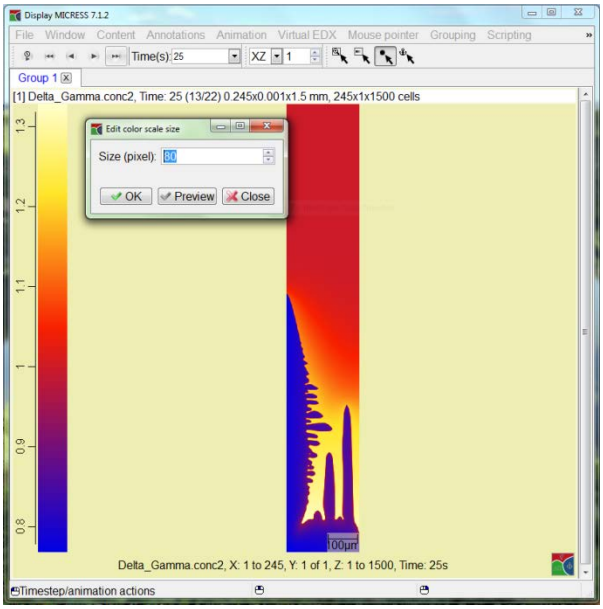

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting
Time(s): 25
XZ 1
Group 1
[1] Delta_Gamma.conc2, Time: 25 (13/22) 0.245x0.001x1.5 mm, 245x1x1500 cells
Edit color scale size
Size (pixel): 80
OK
Preview
Close
100µm
Delta_Gamma.conc2, X: 1 to 245, Y: 1 of 1, Z: 1 to 1500, Time: 25s
Timestep/animation actions

Figure 8.4 Increasing the colour bar thickness

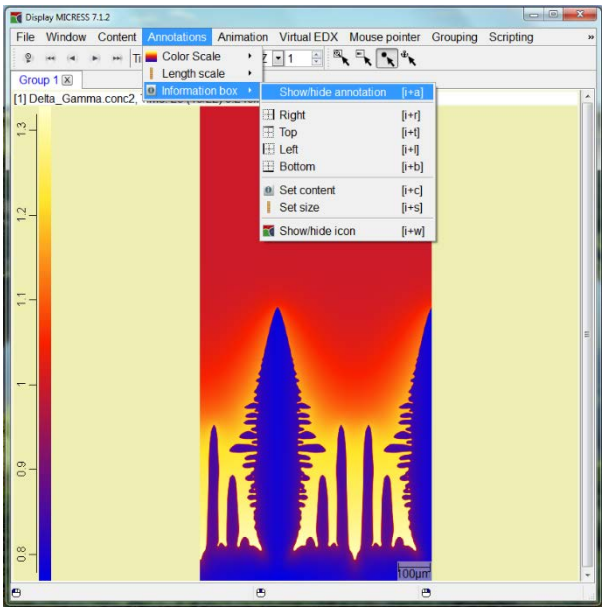


Figure 8.5 Changing to three-fold symmetry

Generate a virtual EDX using the function VirtualEDX → Pick points/lines with mouse → Show line data (See Figure 8.6). With the virtual EDX you can e.g. determine the concentration profile ahead of the dendrite tip, scan concentration across a number of secondary dendrite arms or determine the liquid composition at a point close to the upper boundary. Furthermore, you could add an isoline at a suitable composition in a small distance in front of the dendrites. This can be done using the menu Content → Isoline settings. To display the contours of the isoline, select Normalize and click on 'OK'. Repeat the action to toggle off the isolines and go back to coloured view See Figure 8.7)

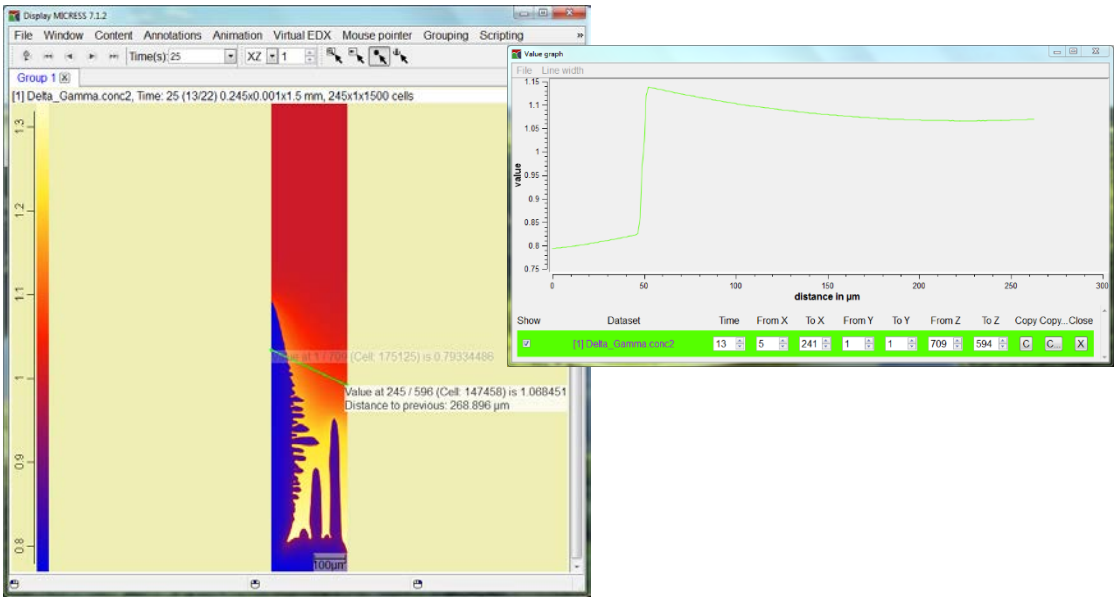

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting
Group 1
[1] Delta_Gamma.conc2, Time: 25 (13/22) 0.245x0.001x1.5 mm, 245x1x1500 cells
Value at 245 / 596 (Cell: 147458) is 1.068451
Distance to previous: 268.896 µm
Value graph
File Line width
value
distance in µm
Show Dataset Time From X To X From Y To Y From Z To Z Copy Copy... Close

Figure 8.6 Generating a virtual EDX

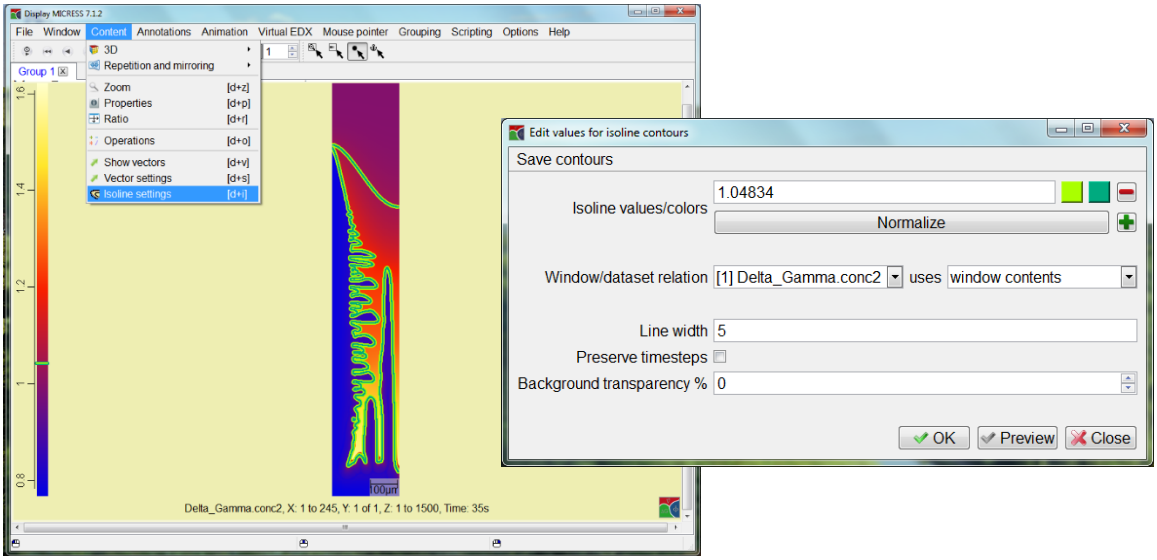


Figure 8.7 Displaying isolines

You can use the option Mouse pointer → Pick points and lines (Virtual EDX to) measure the distance and the angle between two points, to determine the coordinates or the field values of a selected point as well as to obtain extrema and average values in the operating window (see Figure 8.3.).

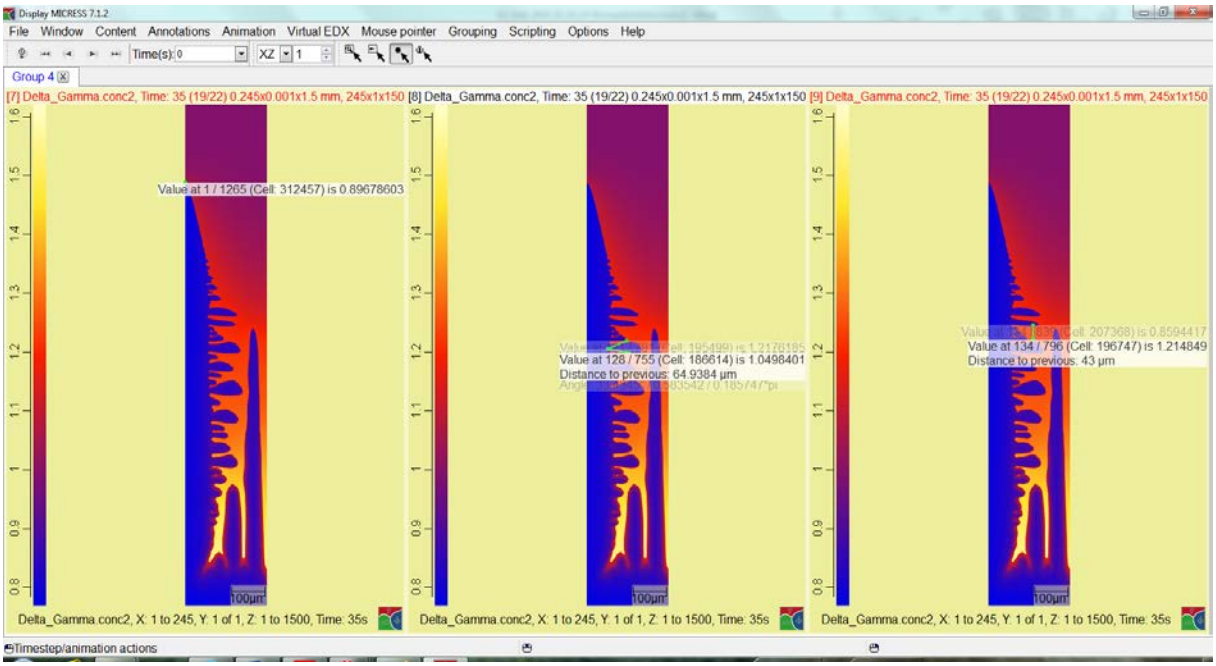

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting Options Help
Group 4
[7] Delta_Gamma.conc2, Time: 35 (19/22) 0.245x0.001x1.5 mm, 245x1x150
[8] Delta_Gamma.conc2, Time: 35 (19/22) 0.245x0.001x1.5 mm, 245x1x150
[9] Delta_Gamma.conc2, Time: 35 (19/22) 0.245x0.001x1.5 mm, 245x1x150
Value at 1 / 1265 (Cell: 312457) is 0.89678603
Value at 128 / 755 (Cell: 186614) is 1.0498401
Distance to previous: 64.9384 µm
Value at 134 / 796 (Cell: 196747) is 1.214849
Distance to previous: 43 µm
Delta_Gamma.conc2, X: 1 to 245, Y: 1 of 1, Z: 1 to 1500, Time: 35s
Timestep/animation actions

Figure 8.8 Extracting numbers with the 'Mouse pointer': Coordinates / field values, angles , distances, extremata and average values. Respective values are displayed in the operating window (left)

8.3 Read and handle a single result file for multiple time steps

When dealing with multiple time steps, one is usually interested in generating different outputs for all time steps. This exercise gives an example how to handle a single output file with multiple time steps. For the purpose, read again the result file for conc2 from the Results_Delta_Gamma folder in your MICRESS installation directory. Before creating the scan, some points should be selected. Then, the scan is generated via Virtual EDX → 'Show line data over time' and consequently the line scan outputs for the selected line are caluclated for all time steps (See Figure 8.9).

If you need virtual EDX outputs for certain time steps, several display scans for specific outputs are selected by clicking on them. Thereby only the curves for the time steps of interest are displayedTo save them, go to File → 'Export as table'.

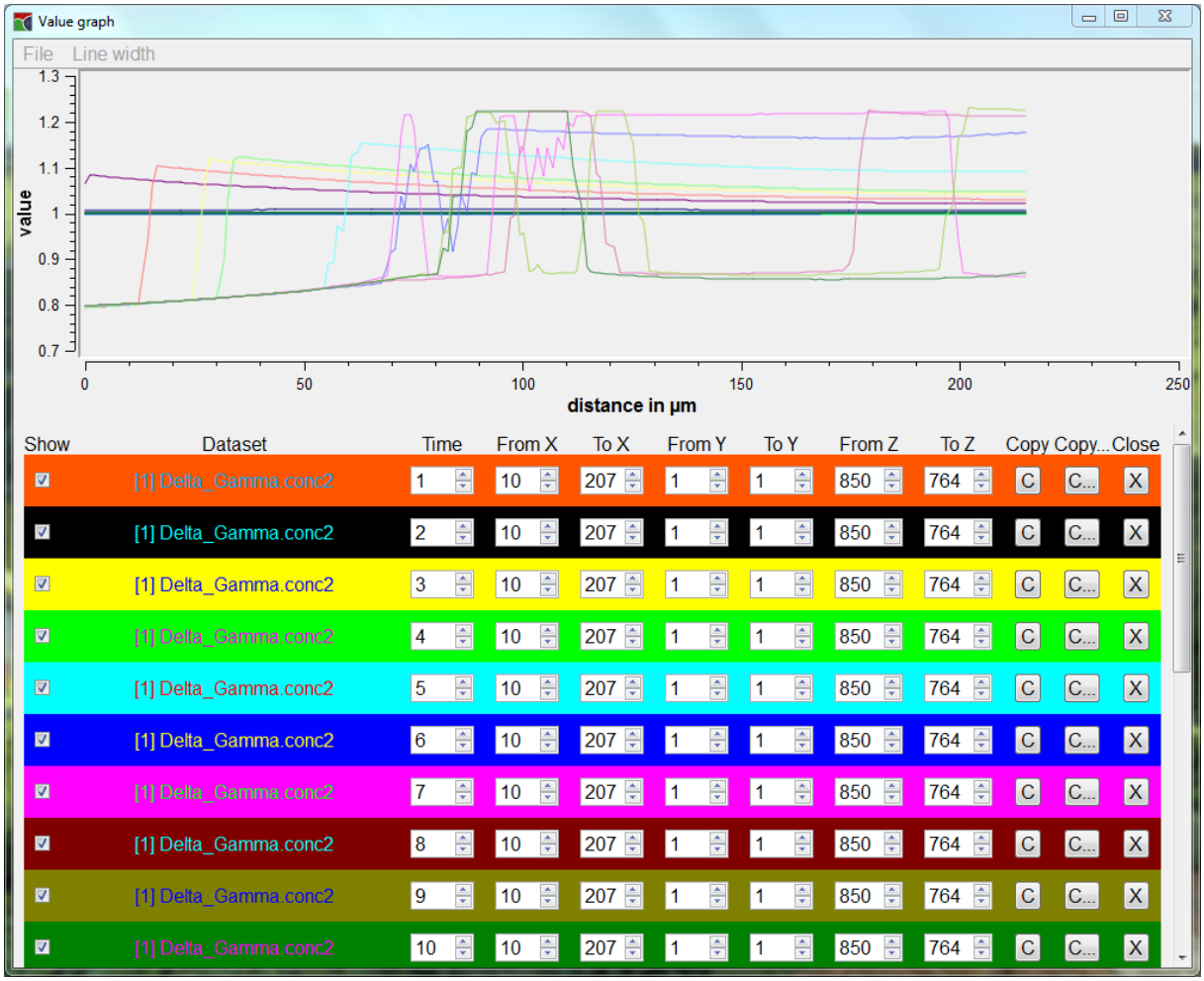

Value graph
File Line width
value
distance in µm
Show Dataset Time From X To X From Y To Y From Z To Z Copy Copy... Close
[1] Delta_Gamma.conc2

Figure 8.9 Generating virtual EDX output for selected time steps

8.4 Read multiple result files

This example demonstrates how to treat multiple result files at a time. First, read simultaneously the result files for the concentration fields of C and Mn and the phase distribution (Delta_Gamma_conc1.mcr, Delta_Gamma_conc2.mcr and Delta_Gamma_phas.mcr). For that purpose, select the three files in the DP_MICRESS window when starting DP_MICRESS. The three result files will be displayed in the same graphic window (see Figure 8.10, left). Hence, zooming in simultaneously on all displayed results can be accomplished as well. For that purpose, use the option 'Mouse pointer' → 'Zoom and move' to zoom in, after having plotted simultaneously all results of interest (See Figure 8.10, right).

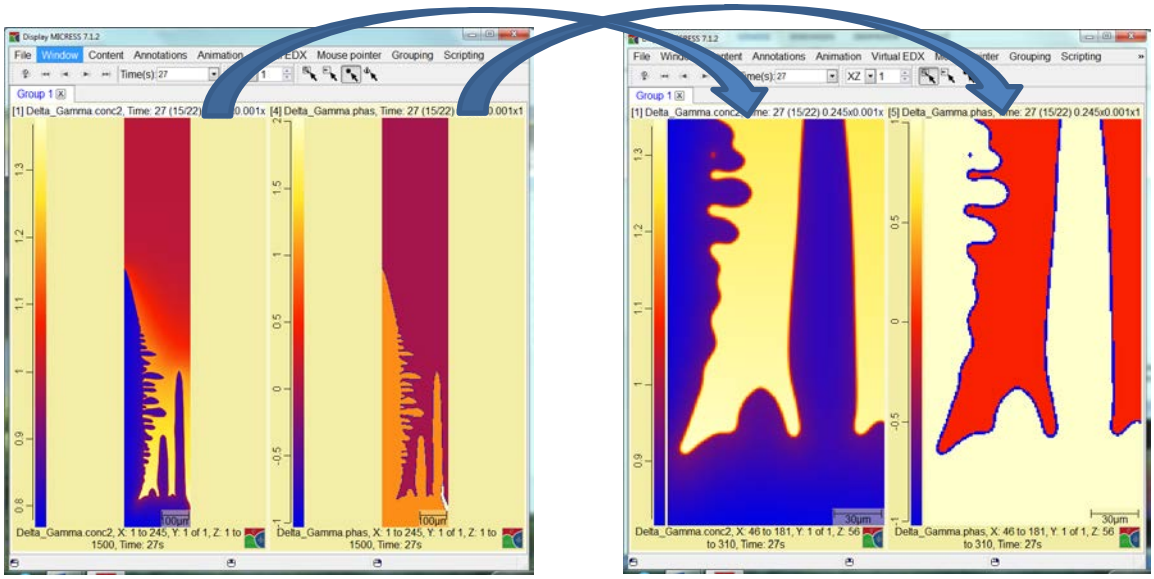


Figure 8.10 Finding the relevant details: Simultaneously zoom in on all displayed results.

In order to have the three panels in a horizontal line, go to 'Window' → 'Tile arrangement' and enter 3 for column count and 1 for row count. Zoom in at all results simultaneously with 'Mouse pointer' → 'Zoom and move'. You only need to select the zoom range on one of the plots and you will automatically zoom in the rest of them.

Next, try to navigate the three plots independent of each other. For that purpose, fix the phase picture at 25 s. This can be done by using the time field to time-step specified by its number. So the activating options can be selected by right clicking on the topline of one dataset. Here, you can select which result files to modify and to which time step to go (see Figure 8.11).

To display the difference between two of the results, you need to have only two results plotted. For that purpose, exit the present plot with the three figures and re-plot only the files Delta_Gamma_conc1.mcr, Delta_Gamma_conc2.mcr. Set the time step for both to be equal e.g. 27 s. Then run DP_MICRESS script in slot 1 'Scripting->Execute script 1: diff'. The difference between the two plots will be displayed in the same graphics window.To display the evolution of a quantity e.g. the carbon concentration in time, copy e.g. Delta_Gamma_conc1.mcr to Delta_Gamma_conc1b.mcr and then read these two files, (original and copy) navigate them in dissociated mode and generate the difference between two different time steps.

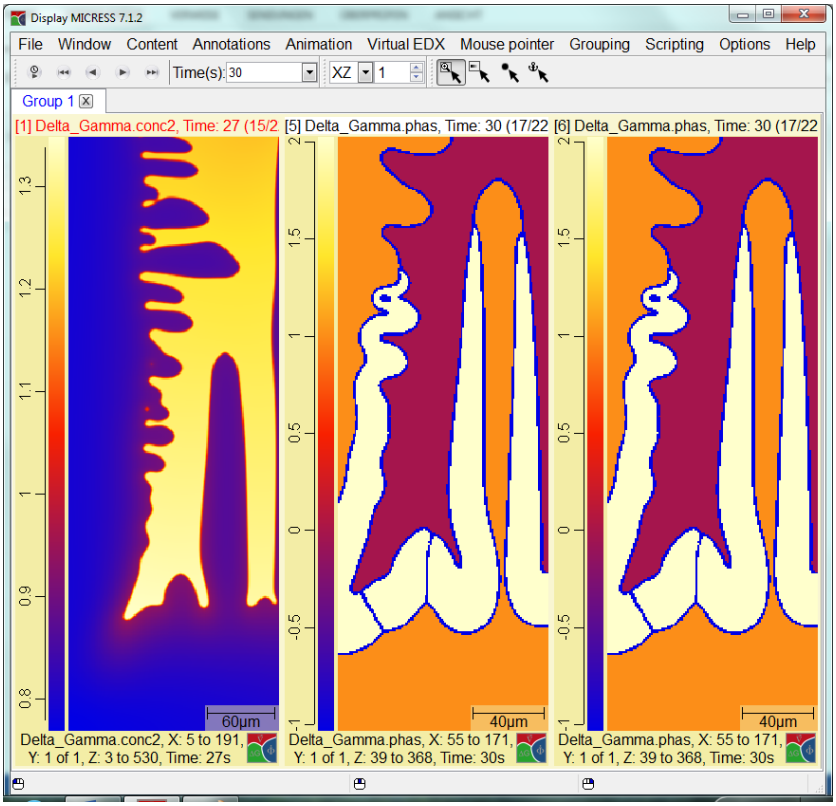

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting Options Help
Time(s): 30
XZ 1
Group 1
[1] Delta_Gamma.conc2, Time: 27 (15/2
[5] Delta_Gamma.phas, Time: 30 (17/22
[6] Delta_Gamma.phas, Time: 30 (17/22
60µm
40µm
40µm
Delta_Gamma.conc2, X: 5 to 191, Y: 1 of 1, Z: 3 to 530, Time: 27s
Delta_Gamma.phas, X: 55 to 171, Y: 1 of 1, Z: 39 to 368, Time: 30s
Delta_Gamma.phas, X: 55 to 171, Y: 1 of 1, Z: 39 to 368, Time: 30s

Figure 8.11 Dissociated navigation of multiple result files plotted simultaneously in DP_MICRESS. The left dataset shows the concentrations of C and Mn at the time step of 27 s, while the figure right shows the (zoomed in) phase distribution at 30 s

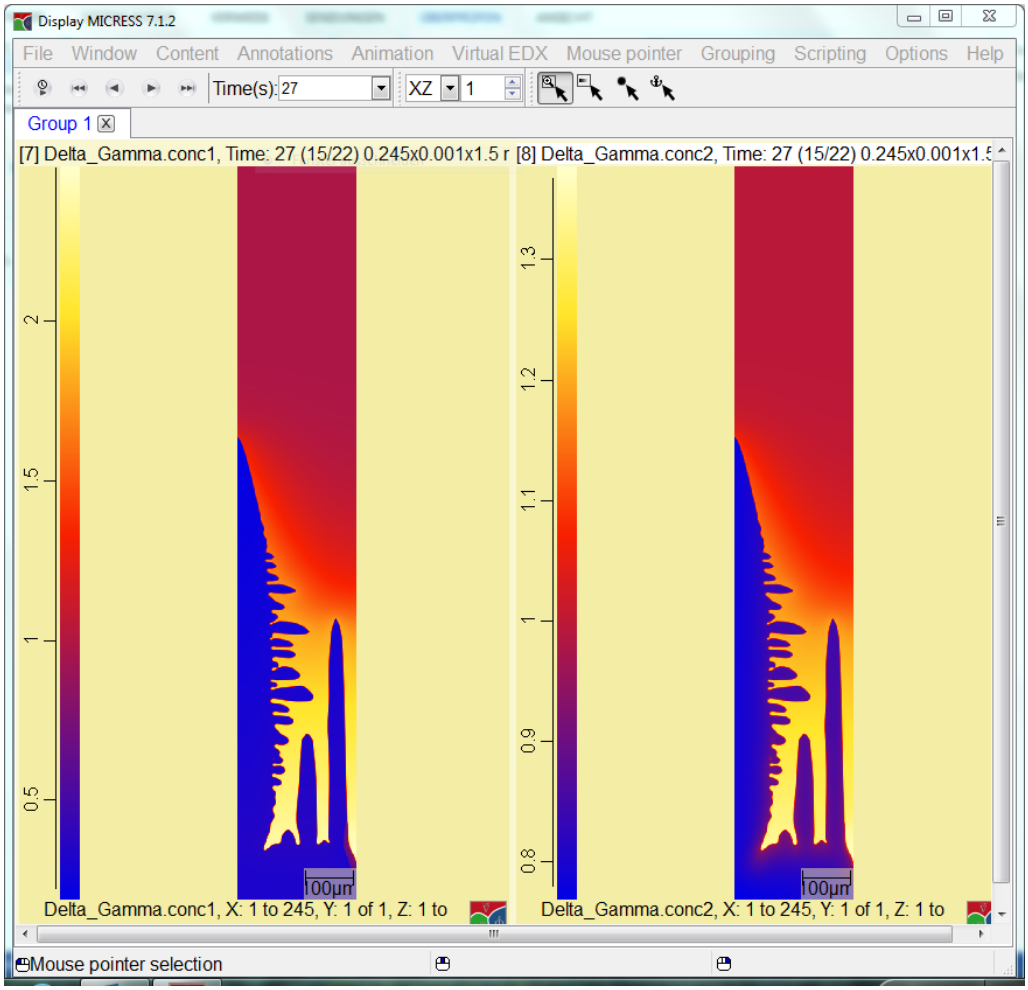

Display MICRESS 7.1.2
File Window Content Annotations Animation Virtual EDX Mouse pointer Grouping Scripting Options Help
Time(s): 27
XZ 1
Group 1
[7] Delta_Gamma.conc1, Time: 27 (15/22) 0.245x0.001x1.5 r
[8] Delta_Gamma.conc2, Time: 27 (15/22) 0.245x0.001x1.5
100µm
100µm
Delta_Gamma.conc1, X: 1 to 245, Y: 1 of 1, Z: 1 to
Delta_Gamma.conc2, X: 1 to 245, Y: 1 of 1, Z: 1 to
Mouse pointer selection

Figure 8.13 Delta_Gamma.conc1 and Delta_Gamma.conc2 at the same time step

9 Useful Information and FAQs

9.1 Why can I not open a certain 3D output file by DP_MICRESS?

The important question whether you can open a MICRESS® result file with DP_MICRESS is the size of the geometry. As the result files are compressed, the shown file size is not relevant. First, the files have to be decompressed before being loaded into the memory. Normally, if MICRESS® is able to calculate and write the results, DP_MICRESS should be able to open them, even it if may take quite a long time.

If you are facing problems with opening a MICRESS® result file with DP_MICRESS, you should first check whether there is any other reason why the file cannot be opened. Such reasons can be limited disk space, limited memory of the computer or too slow operation via a network. If the 32 bit address space is limiting, you should get a corresponding error message.

With DP_MICRESS 7.x, you have the possibility to influence the memory usage of DP_MICRESS (see chapter 4.2.11, Options ->Edit settings->Memory). First method to try is to raise the maximum memory usage for timestep caching to make minimally one timestep fitting in the memory. Second method is to raise the size limit for caching files at all and make DP_MICRESS to read the file each time for disk. This would only be some kind of performant for uncompressed files. Uncompressing can be done easily with every Zip program.

If this is not the case, make a backup of your result files and use DP_MICRESS 6.1 to reduce the size of the file. You can try the following options:

- Use DP_MICRESS 6.1 to transfer the files to vtk-format without displaying them. You can do this either from the graphical interface of DP_MICRESS or in the command line by entering: “DP_MICRESS --VTK newfile.vtk source.phas”. Several source files can be specified, e.g. .conc1 and .phas . Even if you may get the feeling that the procedure does not work, it should produce a correct output. You will probably need a 64bit ParaView version to display the vtk- results.
- Use DP_MICRESS 6.1 to reduce the resolution of the outputs by a factor of 2. Typically, you do not need full resolution for viewing the results. For that purpose, enter “DP_MICRESS --Rescale:1/2 -w newfile source” in the command line. Alternatively, you can use the Graphical Interface (Transfer). Then, you should be able to view the new result files with lower resolution in DP_MICRESS.

If neither of this helps, you can still rerun the MICRESS® simulation and write the outputs directly with smaller resolution. The resolution can be specified in the geometry input of the driving file as a second optional parameter to the grid spacing.

9.2 How to calculate the*.intf-output

The .intf is an integer output. According to the number of combinations of pair wise phase (grain) interactions, it is equal to:

- 1 in dual interfaces
- 3 in triple junctions
- 6 in quadruple junctions

- 10 in quintuple junctions

It is defined in all regions where the fractions are between phMin and 1-phMin, together with the direct neighbour grid cells where one of the fractions is 0 or 1. In the remaining region, the value of .intf is 0.

9.3 How to combine phase boundaries and concentration in DP_MICRESS

The simplest way to combine phase boundaries and concentration outputs in DP_MICRESS would be to draw isolines at a given composition which approximately corresponds to the value in the middle of the phase boundary. This works sufficiently well only if the difference between the concentrations of the present phases is much bigger than the variance inside each phase. In order to draw the isolines, select 'Content->Isoline settings' and configure the desired isoline parameters in DP_MICRESS (shortcut d+i).

A completely different approach would be to use the content operations (see chapter 4.2.3) to create a new result file by combining various MICRESS® outputs. You can e.g. use the fact that phase and grain boundaries are marked as "-1" in the .phas-output and combine this information with a concentration output. For that purpose, use the following approach:

- specifiy a new result window
- select the inputs .phas as 'a' and .conc1 as 'b'.
- specify an operation in 'Muparser' syntax such as: (a == -1) ? 0 : b
- preview or confirm the operation
- export the new dataset as data or image

The new result will be like the original concentration output but with the concentration value 0 along all grain and phase boundaries. Combining different MICRESS® outputs, in general you could obtain any type of graphical visualisation you need. For example, if you would like to have only phase boundaries marked, you can use the phase fraction outputs.

9.4 How to save an animation as a direct output in DP_MICRESS?

There is no direct way to create animations from DP_MICRESS. What you could do is to export all time steps as a series of images (png, tif, jpg, bmp). If you import all these images at once in PowerPoint, they are ordered automatically with respect to the output time. After aligning them all together (e.g. top and right) you can create an animation. This is quite straightforward and fast. Alternatively, you could use a Linux tool like "convert" to make an animated *.gif from the individual images.

10 References

Further information and additional reading can be done at the web-sites of the corresponding freeware for MICRESS® post-processing:

- for DP_MICRESS: www.micress.de
- for ParaView: www.paraview.org
- for gnuplot: www.gnuplot.info

11 Literature

[1] B. Böttger, J. Eiken, I. Steinbach. Phase field simulation of equiaxed solidification in technical alloys. Acta materialia. 2006, Vol. 54, pp. 2697-2704.

[2] B. Böttger, J. Eiken, M.Apel. Phase-field simulation of microstructure formation in technical castings – A self-consistent homoenthalpic approach to the micro–macro problem. J. Comput. Phys. 2009, pp. 6784-6795.

[3] W. T. Read, W. Shockley: “Dislocation models of crystal grain boundaries”, Phys. Rev., vol. 78, n°3, 1950, pp 275-289.

[4] González, G. M. A phase-field model study of two-dimensional dendritic growth in thermal solidification processes“. s.l. : Master Thesis, Supervisor: R. G. Cinca, Departament de Física .

[5] Gupta, S.G. The classical Stefan problem: basic concepts, modelling and analysis. s.l. : Elsevier Science B.N., 2003. 0-444-51085-.

[6] H. Kasajima, E. Nagano, T. Suzuki, S.G. Kim, W.T. Kim. Phase-field Modelling for Faceted Crystal Growth of Silicon. Science and Technology of Advanced Materials. 2003, Vol. 4, pp. 553-57.

[7] H. Meekes, S. X. M. Boerrigter, F. F. A. Hollander, P. Bennema. Needle Crystal Morphology Explained. Chem. Eng. Technol. 2003, Vol. 26, p. 3.

[8] Meirmanov, A. M. The Stefan Problem. De Gruyter Expositions in Math. 3. 1992.

[9] G. Müller, J.-J. Metois and P.Rudolph. Crystal Growth – from Fundamentals to Technology. s.l. : Elsevier B.V., 2004. 0-444-51386-8.

[10] Reed-Hill, R. E. Physical Metallurgy Principles. s.l. : PWS Publishing Company, Boston, 1994. 0-534-98236-0.

[11] Y. Saito, G. Goldbeck-Wood, H. Müller-Krumbhaar. Numerical Simulation of Dendritic Growth. Physical Review A. 1988, Vol. 38, p. 2148.

[12] Schmidt, A. Approximation of Crystalline Dendrite Growth in Two Space Dimensions. Acta Math. Univ. Comenianae. 1998, Vol. LXVII, pp. 57–68.

[13] Stefanescu, D.M. Science and Engineering of casting solidification. s.l. : Kluwer Academic, 2002. 0-306-46750.

[14] Steinbach, I. Phase-field Models in Materials Science. Modelling Simul. Mater. Sci. Eng. 2009, Vol. 17.

[15] Y. Zhang, D. Li, Y. Li. Numerical Simulations of Equiaxed Dendrite Growth Using Phase-field Method. J. Mater. Sci. Technol. 2002, Vol. 18, p. 51.

[16] J. Eiken , B.Böttger, I.Steinbach Multiphase-Field approach for multicomponent alloys with extrapolation scheme for numerical application Phys. Rev. E 73 066122 (2006)

[17] I. Steinbach, F. Pezzolla, B. Nestler, M. Seeßelberg, R. Prieler, G. J. Schmitz, J. L. L. Rezende: A phase field concept for multiphase systems, Physica D 94(1996), p.135-147.

[18] J. Eiken, Numerical solution of the phase-field equation with minimized discretization error, IOP Conf. Ser.: Mater. Sci. Eng. 33 012105, doi:10.1088/1757-899X/33/1/012105.

[19] J. Eiken, The Finite Phase-Field Method - A Numerical Diffuse Interface Approach for Microstructure Simulation with Minimized Discretization Error, MRS Proceedings 2012 1369 : mrss11-1369-xx05-05

DOI: 10.1557/opl.2012.510.

[20] S. G. Fries, B. Böttger, J. Eiken, I. Steinbach. S. G. Fries, B. Böttger, J. Eiken, I. Steinbach:"Upgrading CALPHAD to Microstructure: Simulation: the Phase-field Method. Int. J. Mat. Res. 2002, Vols. 100 – 2.

[21] F.J. Humphreys: "A unified theoy of recovery, recrystallization and grain growth, based on the stability and growth of cellular microstructures – I. the basic model", Acta Mater., vol. 45, n° 10, 1997, pp. 4231-4240.